Farhana N et. al., Vol 2(2), pp 98-116
https://doi.org/10.55306/CJDTES.2025.020203
e-ISSN: 3048-8389
Ci-STEM Journal of Digital Technologies and Expert Systems

Quantum-Dot Cellular Automata-Based Nano-Processors for Next Generation Computing: A Comprehensive Review, Design Methodologies, Performance Analysis and Future Research Directions

Naveed Farhana[1], Humna Abdul Mumeed[2]
[1]Department of Computer Engineering, College of Computers and Information Technology, Taif University, Kingdom of Saudi Arabia.
[2]Faculty of Science, Technology, Engineering and Mathematics (STEM), Open University, United Kingdom
[1]email: d.nfmaqsud@tu.edu.sa, naveedfarhana123@gmail.com. [2]email humna.physics@gmail.com

Open Access
Research Article
Received: 25/05/2025
Accepted: 30/07/2025
Published: 02/09/2025

Corresponding author email:
d.nfmaqsud@tu.edu.sa,
naveedfarhana123@gmail.com.

Citation:
Farhana N et. al., "Quantum-Dot Cellular Automata-Based Nano-Processors for Next-Generation Computing: A Comprehensive Review, Design Methodologies, Performance Analysis, and Future Research Directions" Ci-STEM Journal of Digital Technologies and Expert Systems, Vol. 2(2), pp. 98-116, 2025, doi: 10.55306/CJDTES.2025.020203

Copyright:
©2025 Farhana N et. al.,
This is an open-access article distributed under the terms of the Creative Commons Attribution License which grants the right to use, distribute, and reproduce the material in any medium, provided that proper attribution is given to the original author and source, in accordance with the terms outlined by the license.
(https://creativecommons.org/licenses/by/4.0/)
.

Published By:
Ci-STEM Global Services Foundation, India.

Abstract:
Complementary metal-oxide-semiconductor (CMOS) technology has driven digital electronics for more than five decades, but below the 5 nm node its returns are eroding: short-channel effects, quantum tunnelling leakage and rising power density now consume much of what further scaling delivers. Quantum-dot Cellular Automata (QCA) is among the most actively studied beyond-CMOS alternatives, encoding a bit in the charge configuration of coupled quantum dots rather than in transistor switching current. In principle this allows terahertz-range operation, switching energies far below those of scaled CMOS, and very high integration density. After roughly two decades of circuit-level work, however, QCA processing architectures remain fragmented across isolated combinational and sequential building blocks; a unified, physically consistent and quantitatively benchmarked architecture for a complete QCA nano-processor, covering arithmetic, control, memory, interconnect and clocking, is still missing from the literature. This paper consolidates the physical and mathematical foundations of QCA computation, reviews design contributions published between 2021 and 2025 across adders, multiplexers, memories and arithmetic logic units, proposes a coherent nano-processor architecture built from majority-logic primitives, and benchmarks QCA quantitatively against CMOS and competing beyond-CMOS technologies. The approach combines analytical modelling (Hamiltonian and kink-energy formulations, clocked cell-relaxation dynamics) with architectural synthesis by hierarchical decomposition into fetch/decode, ALU, register-file, memory and clocking subsystems, drawing on a comparative synthesis of fifty-three peer-reviewed sources; normalised projections for area, delay and switching energy are compiled from published simulation benchmarks rather than newly measured. Majority-logic-optimised QCA arithmetic circuits report reductions of roughly 58 to 71 per cent in normalised area-delay product relative to CMOS baselines of comparable generation, with switching energies two to four orders of magnitude below projected sub-5 nm CMOS. Reversible QCA arithmetic logic units report cell-count and area savings near 50 per cent alongside large energy-efficiency gains, though clocking-network overhead, cell-displacement defects and thermal sensitivity remain the unresolved obstacles standing between these results and large-scale integration. On this basis the paper contributes a consistent mathematical treatment of QCA cell dynamics and clocking, a modular nano-processor reference architecture, an updated comparative taxonomy of state-of-the-art QCA building blocks, and a CMOS-QCA benchmarking framework. Priority directions for future work include hybrid CMOS-QCA co-design, molecular and magnetic QCA implementations operating nearer room temperature, defect-aware automated layout synthesis, and machine-learning-assisted clocking-zone optimisation.

Keywords:
Quantum-dot cellular automata; nanoelectronics; beyond-CMOS computing; majority logic gate; QCA clocking scheme; nano-processor architecture; low-power VLSI design; reversible logic; molecular QCA; emerging nanotechnology.

1. INTRODUCTION

For more than five decades the steady downscaling of complementary metal-oxide-semiconductor (CMOS) transistors, described by Moore's law, has been the principal engine of computing performance. That engine is now losing traction. As gate lengths moved into the sub-10 nm and sub-5 nm regimes, several physical barriers arrived at once: short-channel effects, sub-threshold and gate-oxide tunnelling leakage, statistical dopant fluctuation, and a power density that rises faster than performance. Industry roadmaps and academic surveys now treat the search for beyond-CMOS computing substrates as an engineering necessity rather than a speculative exercise.

Among the candidate technologies, which include spintronics, single-electron transistors, carbon-nanotube and two-dimensional-material field-effect transistors, memristive crossbars and quantum computing, Quantum-dot Cellular Automata (QCA) occupies an unusual position because it dispenses with current-driven switching altogether. First proposed in the early 1990s by Lent and co-workers [1], [2], QCA encodes a binary bit as the electrostatic polarisation state of a small number of quantum dots confined within a single cell, rather than as a quantity of charge flowing through a channel. Neighbouring cells interact purely through Coulombic repulsion, so a logic state propagates along an array without any net current [3]. Removing the switching current removes the physical basis for the dissipation that dominates CMOS, and it opens the theoretical prospect of terahertz clock rates at molecular feature sizes.

The computational primitive of QCA is the three-input majority voter, from which the complete Boolean set can be synthesised by fixing one input to a constant polarisation [2]. This foundation differs fundamentally from the pass-transistor and pull-up/pull-down network style of CMOS, and it has motivated a large body of work on QCA-optimised adders, multiplexers, comparators, memory cells and, more recently, complete arithmetic logic units and processor-level architectures [4]–[6].

Despite this volume of circuit-level work, the literature remains fragmented. Individual papers typically optimise a single combinational or sequential block in isolation, minimising cell count, latency or switching energy for one adder or multiplexer topology, without placing the design inside a processor-level architecture that must also handle instruction sequencing, register storage, interconnect wire crossing, clock-zone partitioning and thermal management at scale. A unified treatment bridging cell-level physics, gate-level logic and processor-level organisation is therefore still scarce, particularly one that also situates QCA quantitatively against the CMOS technology it is meant to succeed and against sibling beyond-CMOS proposals.

This paper addresses that gap. It is organised as a review and design study with four interlocking objectives. First, it consolidates the physical and mathematical foundations of QCA computation: cell polarisation, the two-state approximation of the cell Hamiltonian, kink energy, and the four-phase clocking scheme that enforces unidirectional, pipelined information flow. Second, it surveys design contributions published mainly between 2021 and 2025 across combinational arithmetic, reversible and multi-valued logic, memory and ALU design, and synthesises their reported cell count, area, delay and energy figures into a comparative taxonomy. Third, it proposes a modular QCA nano-processor reference architecture integrating a majority-logic ALU, a register file, control and sequencing logic, an interconnect fabric that manages the wire-crossing problem inherent to planar QCA layouts, a reliability-monitoring layer and the clocking network that orchestrates them. Fourth, it develops a quantitative comparison against projected sub-5 nm CMOS and other beyond-CMOS candidates, and closes with a discussion of open reliability and fabrication problems and a set of prioritised research directions.

QCA merits renewed architectural attention for a reason beyond its physical appeal. Over the past decade the beyond-CMOS landscape has converged on a small set of candidate paradigms, each offering a different trade-off among switching energy, integration density, operating temperature and fabrication readiness. Choosing among them requires architectural rather than purely device-level evidence, because a device with excellent isolated figures of merit is of limited practical value unless it can be composed into register files, arithmetic units, memories and control logic without prohibitive overhead from interconnect, clocking and fan-out. QCA is distinctive here in that its most severe integration bottleneck, the planar wire-crossing limitation discussed in Sections 5 and 7, is architectural rather than device-physical in origin. That makes system-level study an unusually high-leverage direction for the field at its current stage of maturity.

The remainder of the paper is organised as follows. Section 2 presents the fundamentals of QCA cell physics, majority logic and clocking. Section 3 reviews the recent literature and positions the present work relative to it. Section 4 develops the mathematical models of cell dynamics, majority-gate behaviour, clocking energy and processor-level performance metrics. Section 5 presents the proposed nano-processor architecture, which builds directly on those models. Section 6 reports the performance evaluation and comparative analysis. Section 7 discusses reliability and fabrication challenges, Section 8 outlines future research directions, and Section 9 concludes.

2. FUNDAMENTALS OF QUANTUM-DOT CELLULAR AUTOMATA

A. The QCA Cell and Binary Polarisation

The elementary building block of QCA is a square cell containing four quantum dots at its corners, populated by two mobile electrons. The electrons tunnel freely between dots within the same cell but are confined from tunnelling to neighbouring cells under normal operating bias [1]. Coulombic repulsion drives the pair to occupy diagonally opposite dots, which yields exactly two energetically degenerate ground-state charge configurations. These are assigned the logic values '1' and '0' and are quantified by a dimensionless cell polarisation P, defined in Section 4-A. Fig. 1 shows the two stable states of an isolated cell, with the occupied dots marked in red. Throughout this paper, occupancy of the upper-left and lower-right dots is taken as P = +1, and the opposite diagonal as P = −1; the same convention is used in Fig. 2 and in (1).

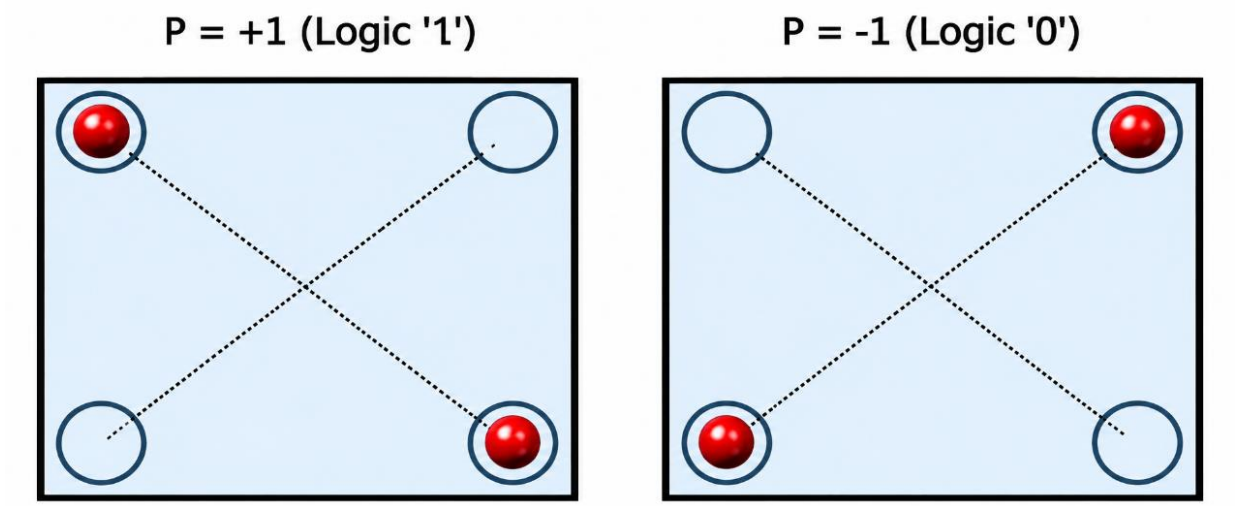


Fig. 1. The two stable polarization states of a single QCA cell: P = +1 (logic '1') and P = −1 (logic '0').

B. Inter-Cell Coupling and Binary Wire Propagation

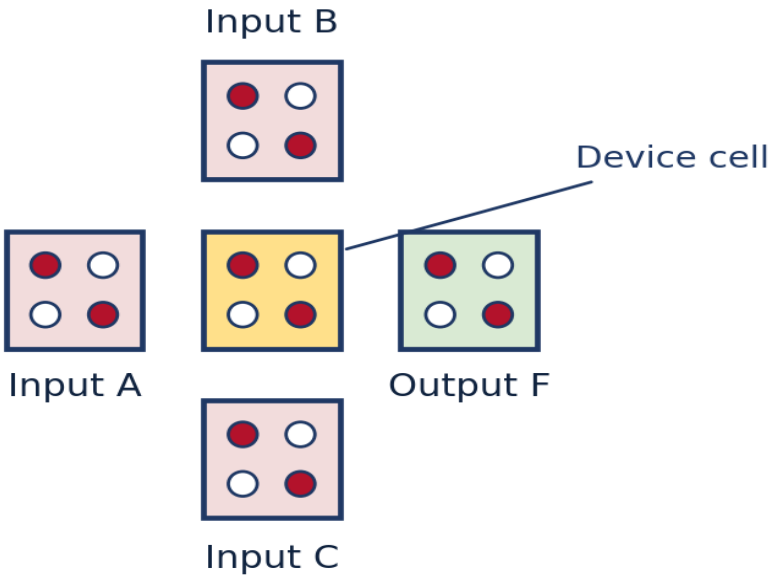
When cells are placed adjacent to one another, the field produced by the electron configuration of one cell shifts the energetically favoured configuration of its neighbour. The polarisation state therefore propagates along a line of cells with no conduction current [2], [3]. A linear chain of cells functions as a binary wire, and a cell rotated by ninety degrees and inserted into a chain realises signal inversion. QCA thus provides both wire propagation and an intrinsic, transistor-free inverter primitive.

C. The Majority Voter as the Universal QCA Primitive

The canonical logic primitive is the three-input majority voter, formed by surrounding a central device cell with three input cells and one output cell, as shown in Fig. 2. The device cell settles into whichever polarisation minimises the total electrostatic energy contributed by its three neighbours, and that state corresponds exactly to the Boolean majority function F(A, B, C) = AB + BC + CA [2]. Fixing any one input to constant logic '0' reduces the gate to a two-input AND; fixing it to constant logic '1' gives a two-input OR. Combined with the cost-free inverter, the majority gate forms a functionally complete basis. Most published QCA combinational circuits, including adders, multiplexers, comparators, encoders and ALUs, are therefore synthesised as networks of majority gates and inverter cells rather than as networks of AND, OR and NOT gates in the CMOS sense.

Because a single majority gate realises only a restricted function of three literals, practical blocks cascade many gates and inverters across several clock zones. The full adder is the standard illustration. The original Tougaw-Lent synthesis uses five majority gates and three inverters per stage [2]; algebraic restructuring of the majority-of-three expressions reduces this to three majority gates and two inverters, and much of the optimisation literature reviewed in Section 3 is concerned with

pushing that count lower still, either through further algebraic manipulation or through direct use of five-input majority gates where the technology supports them [7], [8].



$$F(A, B, C) = AB + BC + CA$$

Fig. 2. Three-input QCA majority voter: F(A, B, C) = AB + BC + CA. All cells are drawn in the P = +1 state of Fig. 1.

D. Clocking and Pipelined Adiabatic Switching

In CMOS a single global clock merely times when combinational outputs are sampled. The QCA clock does something quite different: it is an active, spatially distributed control signal that modulates the inter-dot tunnelling barrier within each cell and so governs whether a cell is free to switch, is holding its state, or is being reset [9], [10]. The widely adopted four-phase scheme partitions the array into clock zones. Each zone cycles through all four phases in turn, and successive zones are offset by one quarter of a period, as illustrated in Fig. 3. During Switch the barrier is raised gradually, letting the cell settle adiabatically into the state dictated by its already-polarised neighbours. During Hold the barrier remains high, latching the state so the cell can act as a stable input to the next zone. During Release and Relax the barrier is lowered again, erasing the polarisation in preparation for the next computational wave.

This staged, adiabatic switching is the source of the very low dissipation attributed to QCA, because the cell is never forced to switch against an opposing field, in contrast to the abrupt charging and discharging of a CMOS gate capacitance [11]. The clocking network simultaneously provides the pipelining mechanism that lets successive data words occupy successive clock zones, which gives QCA circuits an inherent wave-pipelined throughput advantage [12].

One consequence of clock-zone partitioning is frequently underemphasised in circuit-level papers. The number of zones spanned by the longest combinational path sets the pipeline latency of a block in clock cycles, independently of its raw cell count. Two adders with identical cell counts can therefore differ markedly in latency if one is laid out with a shallower zone depth. This trade-off is a principal motivation for alternative clocking schemes such as USE (Universal, Scalable and Efficient) [13] and one-dimensional clocking [14], both of which aim to reduce zone-boundary overhead relative to the regular four-zone rotation used in most early designs. Recent reviews of regular clocking schemes compare these options systematically [15].

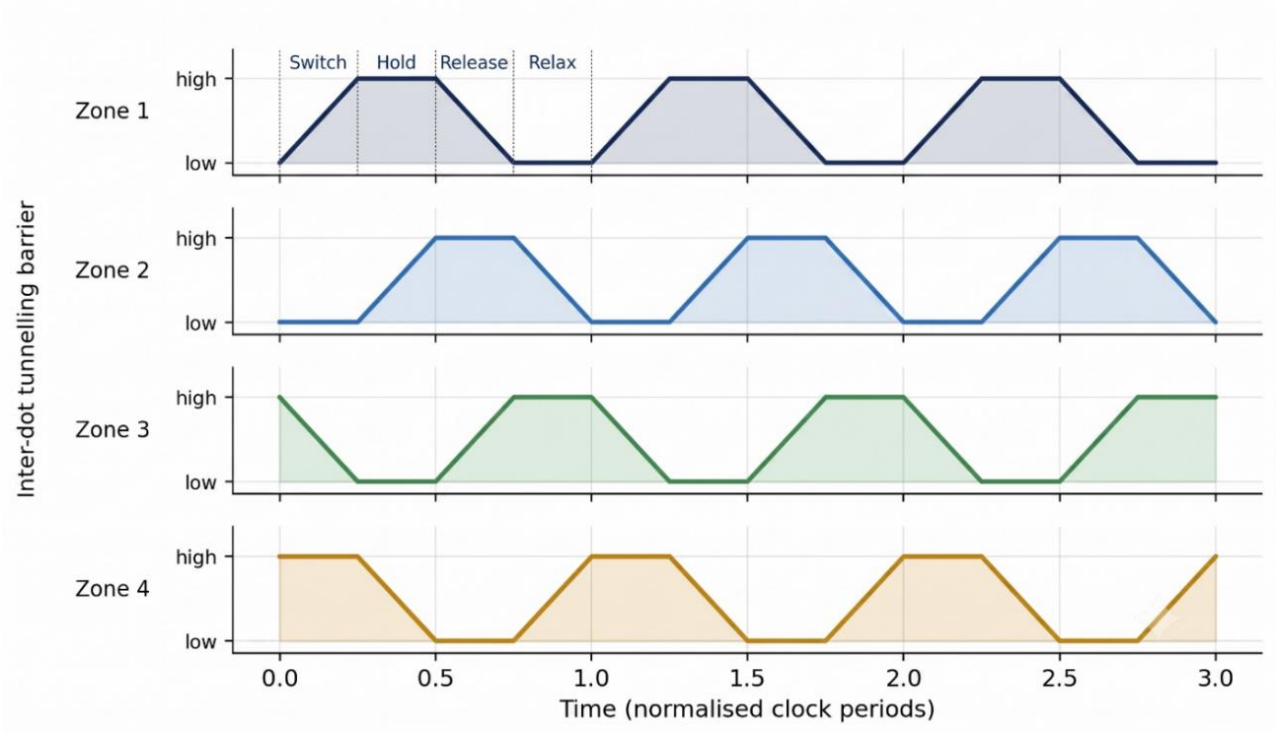


Fig. 3. Four-phase QCA clocking. Every zone traverses Switch, Hold, Release and Relax; successive zones are offset by one quarter period.

E. Design and Verification Methodology

QCA circuit design in the reviewed literature, and in the architecture proposed in Section 5, follows a broadly consistent three-stage methodology. In the first stage a target Boolean or arithmetic function is expressed as a network of majority-of-three or, less commonly, majority-of-five expressions using standard majority-logic identities. This mirrors the role technology mapping plays in CMOS standard-cell synthesis. In the second stage the gate network is mapped onto a physical cell layout and partitioned into clock zones, typically using a bistable or coherence-vector simulator such as QCADesigner [16], [17], to confirm that the intended logic function survives once realistic cell-to-cell coupling and clocking delay are taken into account. Wire-crossing decisions between coplanar and multilayer routing are also made at this stage [18]. In the third stage the verified layout is evaluated against cell count, area, latency in clock cycles and switching energy, the metrics that dominate the comparative tables of Section 6. Where reliability is a design objective, the layout is additionally subjected to defect injection, in which individual cells are randomly omitted, displaced or mis-polarised to estimate functional yield under realistic fabrication tolerances [19], [20]. Energy figures are commonly obtained with QCAPro or QCADesigner-E using the upper-bound dissipation model of Srivastava and co-workers [21], [22]. The same three-stage methodology underlies the architectural subsystems proposed in Section 5 and the figures compiled in Section 6.

3. LITERATURE REVIEW AND RELATED WORK

The theoretical foundation of QCA was laid by Lent and co-workers, who formulated the cell as a coupled quantum-dot system and derived its Hamiltonian and switching dynamics [1]–[3]. Tóth and Lent extended the treatment to quasi-adiabatic switching in metal-island implementations [9], and Timler and Lent established the power-gain and dissipation relations on which nearly all later energy analysis rests [11]. Simulation tooling matured with the QCA Designer family of bistable and coherence-vector simulators [16], [17], which made systematic exploration of cell placement and clock-zone assignment practical and which remains the de facto standard verification environment across the field. Power estimation was placed on a firmer footing by the upper-bound dissipation model of Srivastava, Sarkar and Bhanja [21] and the QCA Pro tool built on it [22].

A substantial recent literature focuses on optimising arithmetic building blocks. Area- and defect-aware adder and subtractor structures have been reported that reduce cell count while explicitly

analysing robustness to fabrication-level cell displacement [23], and XOR-gate-based adder implementations with physical-level verification have shown competitive latency relative to majority-gate-only designs [24]. Work published in 2025 has continued along the same line: optimised nanoscale adder-subtractor architectures report cell-complexity reductions of five, twenty-seven and eleven per cent for the adder, subtractor and combined adder-subtractor respectively against prior designs [8], while an area-delay efficient binary-coded-decimal adder extends the approach to decimal arithmetic [25]. Comparator circuits have been optimised for reduced cell count, latency and area [26], and multiplexer design continues to attract attention, with low-cell-count 2:1 and higher-order topologies [27], [28] reported alongside fault-tolerant variants that trade a modest area overhead for resilience to single-cell omission and mis-polarisation [29]. Demultiplexer designs with explicit energy estimation complete the routing-primitive picture [30].

Memory design forms a second major thread, since a practical processor needs on-chip storage with area and latency compatible with its logic fabric. Loop-based QCA-RAM cells using asynchronous set and reset around a multiplexer and latch reduce footprint and cost substantially relative to earlier proposals [31], [32], and three-layer multilayer organisations have been proposed specifically to improve scalability for larger m × n arrays [31]. Vertical integration is increasingly seen as necessary to overcome the wire-crossing area penalty of purely coplanar layouts, a view reinforced by 2025 work on multilayer programmable logic blocks [33] and on multilayer molecular QCA multiplexing structures applied to RAM cell construction [34]. A 2025 study integrating RAM, demultiplexer and ALU into a single design flow reports reductions of sixty-seven per cent in cell count, sixty-six per cent in area and roughly ninety-two per cent in quantum cost for its RAM cell relative to earlier designs, together with a twenty per cent latency reduction [6]. Clocking research has advanced in parallel: reviews of regular clocking approaches synthesise the trade-offs between fixed four-zone, USE and one-dimensional schemes [15], and low-latency ALU designs exploiting one-dimensional clocking report reduced critical-path delay relative to conventional four-zone implementations [14].

A third and increasingly active thread concerns reversible and multi-valued logic, motivated by the Landauer bound on the energy dissipated by irreversible bit erasure [35]. Reversible sequential elements, including physically reversible flip-flops, have been reported with markedly reduced dissipation relative to irreversible counterparts [36]–[38], and level-sensitive T flip-flops built from compact XOR/XNOR gate pairs have been proposed to lower the cell cost of the sequential layer itself [39]. The reversible ALU of Edwards and co-workers is the most quantitatively explicit of these: measured against the next most efficient sixteen-operation QCA ALU, it uses fifty-one per cent fewer cells and forty-seven per cent less area while improving energy efficiency by 88.8 per cent, and it pursues physical rather than merely logical reversibility [40]. A companion reversible 8:1 multiplexer applies the same design philosophy to routing [41]. Parallel work on ternary and multi-valued QCA has extended reversible methodology beyond binary logic, with ternary Toffoli and Feynman gate structures [42] built on multi-valued cell models and two-layer synchronised ternary wire crossings [43], motivated by the higher information density available per cell under a multi-valued encoding.

Reliability, thermal behaviour and fabrication feasibility have received growing scrutiny as the field matures beyond purely logical simulation. Studies of fault tolerance under regular clocking have quantified how cell-deposition and displacement defects propagate through majority-gate networks [20], and fault-tolerant majority voters and ALUs have been designed specifically to contain that propagation [7]. Thermal work published in 2025 addresses the temperature sensitivity of polarisation directly: a thermally stable coplanar four-bit carry-save adder demonstrates that careful cell-size scaling can widen the usable temperature window [44], while reliability-focused nanoscale circuit designs report stable average output polarisation across roughly 1 to 8 K [45]. Molecular QCA work, including analysis of electric-field inputs [46] and of clock topologies for molecular implementations [47], points toward a room-temperature-viable pathway that is physically distinct from the metal-island and semiconductor-dot implementations studied in most simulation-only papers. Complementary work on programmable crossbar QCA circuits has explored FPGA-like reconfigurable fabrics as a route to design-automation-friendly large-scale systems [48].

At the processor-integration level the literature is comparatively thin. Fazzion and co-workers described a complete SUBNEG-class QCA processor, establishing that full fetch-decode-execute organisation is achievable in QCA at proof-of-concept scale [4]. Subsequent ALU-focused studies have proposed ultra-area-efficient designs using synchronised clock-zone scheduling [5], multilayer single-bit ALUs with reduced cell count [49], and ALUs targeted at Internet-of-Things workloads [50]. A low-latency, area-efficient design aimed explicitly at next-generation sustainable digital systems extends the same line of work [51]. What remains rare is a proposal that integrates an ALU, register file, memory bank, interconnect fabric and clocking network into a single quantitatively benchmarked architecture. Narrowing that gap is the purpose of Section 5.

Design automation constitutes a final, cross-cutting thread. Walus and Jullien framed the absence of automated synthesis and place-and-route flows as the primary obstacle to circuit-scale rather than gate-scale QCA design [52]. That assessment remains broadly valid: most of the adder, multiplexer and ALU papers surveyed above still report largely manual or semi-automated layout. Part of the difficulty is intrinsic, since placement and routing for tile-based field-coupled nanocomputing circuits has been shown to be NP-complete [53]. Coplanar crossover techniques that exploit clock-zone timing differences to route two logically independent wires across the same physical layer offer a lower-fabrication-complexity alternative to true multilayer routing [18], and the trade-off between the two bears directly on the interconnect choice made in Section 5.

Table 1 summarises the principal research threads identified in this review, their representative focus and the specific gap each leaves relative to a complete, benchmarked nano-processor architecture.

Table 1. Principal QCA Research Threads and the Gaps Addressed by This Paper

Research thread	Representative focus	Gap relative to a complete nano-processor
Arithmetic circuit optimisation	Cell-count and area-reduced adders, comparators, multiplexers [8], [23]–[29]	Blocks optimised in isolation, not integrated with control or memory
Memory design	Loop-based and multilayer QCA-RAM [6], [31], [32], [34]	Memory-ALU-register interconnection rarely modelled
Clocking schemes	Four-zone, USE and one-dimensional trade-offs [12]–[15]	No agreed processor-scale clocking hierarchy
Reversible and multi-valued logic	Reversible flip-flops and ALUs; ternary gates [36]–[38], [40]–[43]	Demonstrated at block level, not at datapath scale
Reliability and thermal analysis	Defect propagation and thermal sensitivity [19], [20], [44], [45]	Rarely integrated as an architectural monitoring layer
Design automation and tooling	QCADesigner, crossbar and coplanar crossover [16]–[18], [48], [52], [53]	Manual layout still dominant; limited synthesis automation

4. MATHEMATICAL MODELLING AND THEORETICAL ANALYSIS

The architecture proposed in Section 5 rests on a chain of analytical relations that runs from the single-cell description, through inter-cell coupling and majority-gate energetics, to clocked switching energy and, finally, the processor-level metrics used in Section 6. That chain is developed here.

A. Two-State Approximation and Cell Polarisation

Under the widely used two-state approximation, each cell is modelled as a two-level quantum system whose state is fully described by its polarisation P. Numbering the dots clockwise from the upper-left corner, so that dots 1 and 3 lie on the upper-left/lower-right diagonal and dots 2 and 4 on the other, and writing ρ_1 to ρ_4 for the corresponding dot-occupation probabilities, the polarisation is

$$P = \frac{(\rho_1+\rho_3)-(\rho_2+\rho_4)}{\rho_1+\rho_2+\rho_3+\rho_4} \quad (1)$$

This evaluates to P = +1 for the diagonal occupation assigned logic '1' in Fig. 1 and P = −1 for the orthogonal diagonal assigned logic '0'. Intermediate values −1 < P < 1 correspond to a quantum-mechanically indeterminate or thermally perturbed state. They matter for reliability analysis, because a cell's effective logic margin can be quantified by how tightly its steady-state polarisation clusters near ±1 under given noise and temperature conditions [45].

B. Cell Hamiltonian and Kink Energy

The two-state Hamiltonian of a cell embedded in a neighbourhood of already-polarised cells is conventionally written in the Pauli basis as

$$H = -\gamma\sigma_x - \frac{1}{2}\sigma_z \sum_j f_{ij} E_k P_j \quad (2)$$

Here γ is the tunnelling energy between the two dot-pair basis states, the term responsible for quantum superposition and hence for adiabatic switching, and E_k is the electrostatic kink energy associated with a neighbouring cell j at polarisation P_j. The geometric factor f_{ij} captures how the interaction weakens with separation. Because a polarised QCA cell carries no net charge and no net dipole moment, the leading electrostatic interaction between two cells is quadrupole-quadrupole, which falls off as the inverse fifth power of the inter-cell distance [3]. This steep decay is what makes the nearest-neighbour approximation used throughout the layout literature reasonable, and it is also why cell spacing is such a sensitive design parameter.

The kink energy is the energy penalty of a polarisation discontinuity, or kink, along an otherwise uniformly polarised wire. It is obtained by summing the Coulomb interaction over all charge pairs in the two cells and taking the difference between the aligned and anti-aligned configurations:

$$E_k = \frac{1}{4\pi\varepsilon_0\varepsilon_r} \sum_i \sum_j q_i q_j \left[\frac{1}{d_{ij}^{anti}} - \frac{1}{d_{ij}^{aligned}} \right] \quad (3)$$

where q_i and q_j are the charges at dot i of the first cell and dot j of the second, d_{ij} is the separation between them evaluated in the anti-aligned and aligned configurations respectively, and $\varepsilon_0\varepsilon_r$ is the permittivity of the surrounding medium. Written this way the expression is dimensionally consistent, each term having units of energy, and it reduces to the familiar four-term difference for the idealised two-electron cell. E_k sets the fundamental energy scale that must exceed the ambient thermal energy k^BT by a comfortable margin for reliable binary operation. That requirement is precisely why the maximum viable operating temperature, and ultimately the choice between semiconductor-dot, metal-island and molecular implementations, is dictated by how large a kink energy the physical realisation can achieve. Because E_k scales inversely with inter-dot separation, molecular implementations with sub-nanometre spacing are theoretically capable of kink energies large enough to sustain room-temperature operation [46], [47]. This is the central motivation for the molecular-QCA direction discussed in Sections 3 and 8.

C. Majority Gate Energetics

For the three-input majority gate of Fig. 2, the device-cell polarisation P^d that minimises total system energy can be shown, in the strong-coupling adiabatic limit, to follow

$$P_d = sign(\alpha_A P_A + \alpha_B P_B + \alpha_C P_C) \quad (4)$$

where P_a, P^b and P_c are the fully polarised input values and the coefficients α_a, α^b and α_c are proportional to the respective inter-cell kink-energy coupling strengths, normalised so that they sum to unity. For three geometrically symmetric inputs the coefficients are equal and (4) reduces to sign($P_a + P^b + P_c$), recovering the Boolean relation F = AB + BC + CA used throughout the synthesis literature. Asymmetric input placement, which arises routinely in dense layouts, unbalances the coefficients and shifts the effective decision threshold. This is a documented source of majority-gate misbehaviour that layout-aware tools must guard against explicitly [7], [20].

D. Clocked Switching Energy and the Landauer Bound

During the Switch phase the tunnelling barrier γ(t) is raised quasi-statically from γ_{min} to γ_{max} over a switching interval τ_s set by the clock. Under adiabatic switching the dissipated energy per cell switching event can be approximated by

$$E_e \approx \frac{k_d(\gamma_{max} - \gamma_{min})^2}{\tau_s} \quad (5)$$

where k^d is a technology-dependent damping coefficient. Because E_e scales inversely with τ_s, adiabatic QCA switching trades speed against energy in a way that is qualitatively different from CMOS dynamic power, which follows $P = \alpha CV^2 f$ and is independent of how slowly an individual transition occurs. This inverse-τ_s scaling is the formal basis for the low switching-energy figures reported throughout the QCA literature at sub-terahertz to low-terahertz clocking, and for the sub-attojoule per-cell values projected for molecular implementations [11], [21], [35]. Section 6 contrasts these values with the Landauer minimum for irreversible bit erasure, $k^B T \ln 2$.

For circuits synthesised from logically reversible gate networks, such as the reversible ALU and flip-flop designs discussed in Section 3, no net information is erased during normal operation. The dissipation is then bounded in principle only by the non-adiabatic component of the switching process rather than by the Landauer bound. That is the mechanism behind the large energy-efficiency improvements reported for reversible QCA circuits, of which the 88.8 per cent figure quoted for the reversible ALU of [40] is representative. It should be read as a relative improvement over an irreversible design of equal logical function rather than as an absolute energy figure.

E. Processor-Level Performance Metrics

To evaluate the architecture of Section 5 on the same footing as CMOS processor design practice, three composite metrics are adopted. The area-delay product is computed as $ADP = A \cdot D$, where A is total occupied cell area and D is critical-path latency expressed in clock-zone cycles. This is the quantity plotted in normalised form in Fig. 5 and tabulated in Table 3. The energy-delay product combines the switching-energy model of Section 4-D with the same latency term,

$$EDP = E_e \cdot D \tag{6}$$

and is the more informative measure when comparing reversible against irreversible ALU configurations, because the two trade cell count, and hence area, against dissipated energy in opposite directions. Finally, sustained instruction throughput T for the wave-pipelined execution model introduced in Section 5 is approximated by

$$T \approx \frac{f_{clock}}{Z_{min}} \tag{7}$$

where f_{clock} is the clock-zone switching frequency and Z_{min} is the minimum clock-zone depth among the pipeline's constituent subsystems. Equation (7) makes an important architectural point explicit. Unlike single-cycle CMOS throughput, which is limited by the longest combinational path, wave-pipelined QCA throughput is limited by the shallowest stage, because that stage determines how frequently a new computational wave can be admitted. Balancing zone depth across the ALU, register file and memory subsystems therefore matters more than minimising any one of them, a point returned to in Section 5.

5. PROPOSED QCA-BASED NANO-PROCESSOR ARCHITECTURE

Building on the circuit-level advances surveyed in Section 3 and the models developed in Section 4, this section proposes a modular reference architecture that integrates the subsystems required for general-purpose computation into a coherent, clock-synchronised whole. The architecture is shown in Fig. 4. It follows a conventional functional decomposition into fetch and decode, register file, arithmetic logic unit, memory, control and interconnect, but every block is reconceived as a majority-logic network governed by a shared four-phase clocking hierarchy rather than as a transistor-level datapath.

At the top of the hierarchy a QCA clock generator produces the four-phase zoning signal distributed to every downstream subsystem, shown by the dashed gold paths in Fig. 4. Because QCA computation is inherently wave-pipelined, the zoning topology directly determines achievable instruction throughput through (7). The architecture adopts a hybrid strategy: regular four-zone clocking within tightly coupled arithmetic blocks, where latency predictability matters most, and a one-dimensional clocking variant along long interconnect runs, where its reduced zone-boundary overhead is an advantage [14], [15].

The instruction fetch and decode unit and the control and sequencing logic interpret a QCA-compatible instruction encoding and generate the control signals that select ALU operations, register-file ports and memory addressing. The register file is realised as an array of loop-based

latches, the same cross-coupled majority-gate feedback structure used in the memory bank, organised for single-cycle read access consistent with its position on the critical path of every arithmetic instruction [6], [31].

The arithmetic logic unit is the computational core and is synthesised entirely from majority-gate primitives. A carry-optimised adder network forms the basis of addition and subtraction, the latter via two's-complement input inversion, while multiplication uses a majority-gate partial-product array analogous in structure, though not in switching physics, to a CMOS array multiplier [8], [23]. Consistent with the reversible-logic literature reviewed in Section 3, the ALU admits an optional reversible configuration in which Toffoli- and Feynman-equivalent majority-gate constructions replace the standard irreversible adder cells [40], [42]. This trades a modest increase in cell count for a substantial reduction in dissipation, a trade-off made explicit in Section 6.

The memory bank uses loop-based QCA-RAM cells built from cross-coupled 2:1 multiplexer and D-latch structures with asynchronous set and reset, following the multilayer organisation shown to improve footprint scalability for m × n arrays [31], [32]. Because a purely coplanar layout cannot cross two independent binary wires without merging their polarisation states, the architecture relies on a dedicated interconnect fabric. Short same-layer connections use coplanar routing, including clock-zone-based crossovers where timing permits [18], while long-haul buses between the ALU, register file and memory bank use multilayer crossbar routing, mirroring the approaches proposed for reconfigurable QCA fabrics [33], [48].

Because polarisation fidelity is sensitive to thermal noise and to cell-placement defects, the architecture incorporates a thermal and reliability monitoring layer, conceived as a distributed network of low-overhead majority-gate parity and redundancy checkers inserted at subsystem boundaries [7], [20]. This layer does not correct errors in the strict sense of triple-modular redundancy, which would impose a severe area penalty. Instead it flags subsystems whose output polarisation statistics deviate from expected margins, enabling graceful degradation or re-execution rather than silent data corruption. An I/O interface and mixed-signal CMOS-QCA bridge terminates the architecture, reflecting the near-term reality that QCA nano-processors are most plausibly deployed as hybrid co-integrated accelerators rather than as fully standalone chips.

Instruction execution proceeds as a clock-zone-synchronised wave rather than as a single-cycle CMOS-style datapath traversal. A fetched instruction word propagates from the fetch and decode unit through the control logic, register file, ALU and, where required, the memory bank and interconnect fabric, with each subsystem boundary corresponding to a small integer number of clock zones. Successive instructions can be admitted at a rate governed by (7), so steady-state throughput depends on the shallowest stage rather than on the end-to-end latency of a single instruction. The latter remains the relevant figure for single-instruction latency, but not for sustained throughput.

This execution model has two consequences that distinguish the proposal from a naive assembly of previously published components. First, subsystem boundaries must be zone-depth balanced during integration. If the ALU spans substantially more clock zones than the register-file read path, the mismatch creates a pipeline bubble that erodes the very throughput advantage wave pipelining is meant to provide. The zone allocations in Table 2 are therefore chosen to be depth-matched rather than independently minimised. Second, the interconnect fabric's choice between coplanar and multilayer routing must itself be zone-aware, since a coplanar clock-zone-based crossing consumes additional zones along the crossing path relative to a multilayer crossing. This creates an explicit area-versus-latency trade-off at the interconnect level that mirrors, at processor scale, the area-versus-fabrication-complexity trade-off already noted at block level in Section 3.

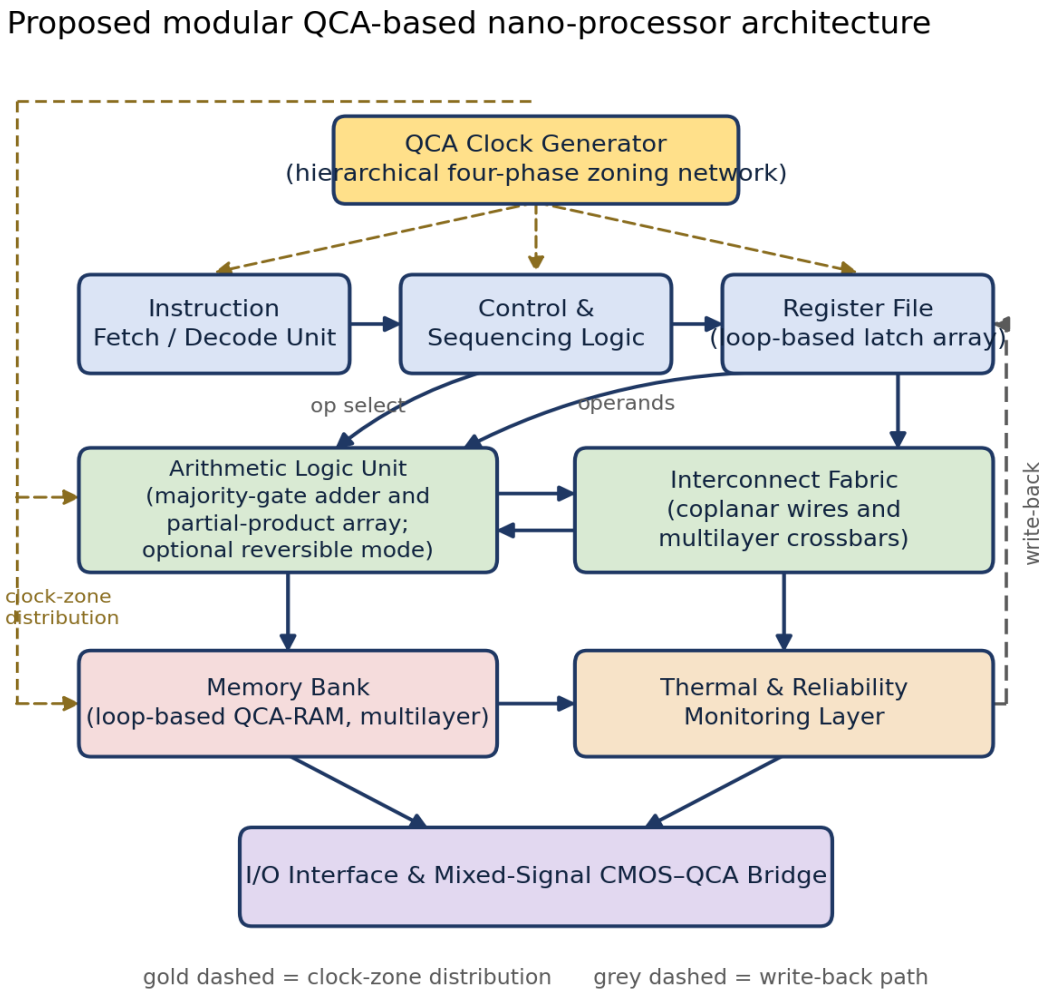


Fig. 4. Proposed modular QCA-based nano-processor reference architecture.

Table 2 summarises the functional role, majority-gate synthesis basis and clocking strategy of each subsystem.

Table 2. Functional Subsystems of the Proposed QCA Nano-Processor

Subsystem	Functional role	Majority-logic synthesis basis	Clocking strategy
Clock generator	Distributes the four-phase zoning signal to all subsystems	Control layer; not a logic network	Global hierarchical zoning
Fetch/decode and control	Instruction interpretation and control-signal generation	Majority-gate decoder and sequencer network	Four-zone regular
Register file	Fast operand storage for ALU operands	Loop-based latch array	Four-zone regular
Arithmetic logic unit	Addition, subtraction, multiplication; optional reversible mode	Majority-gate adder and partial-product array	Four-zone regular
Memory bank	Program and data storage (QCA-RAM)	Multiplexer and D-latch loop cells, multilayer	Four-zone regular

Subsystem	Functional role	Majority-logic synthesis basis	Clocking strategy
Interconnect fabric	Inter-subsystem bus routing and wire crossing	Coplanar wires and multilayer crossbars	One-dimensional on long runs
Reliability monitor	Parity and redundancy-based fault flagging	Lightweight majority-gate parity checkers	Four-zone regular
CMOS-QCA bridge	I/O and mixed-signal interfacing	Hybrid boundary, outside the QCA domain	External CMOS clock domain

6. PERFORMANCE EVALUATION AND COMPARATIVE ANALYSIS

To place the proposed architecture and the surveyed literature in a common quantitative frame, this section compiles normalised figures for QCA against CMOS at the circuit-block level, for QCA against sibling beyond-CMOS technologies at the qualitative level, and for switching-energy trends across technology generations. All quantitative figures are compiled and normalised from the cited published sources rather than newly measured. They are presented as representative trends, not as claims of new experimental results, and the normalisation makes them comparable to one another rather than absolute.

Table 3 summarises reported cell-count, area and delay figures for representative QCA implementations of standard arithmetic blocks relative to CMOS baselines drawn from the same publication class. Fig. 5 plots the resulting normalised area-delay product across four block types: a one-bit full adder, a four-bit ripple-carry adder, an 8 × 8 multiplier and a four-bit ALU. Across these blocks the majority-gate-optimised QCA designs show reductions in normalised area-delay product of roughly 58 to 71 per cent relative to their CMOS counterparts at comparable technology-generation assumptions. The largest relative gain appears for the multiplier, where cascaded majority gates avoid the transistor-count growth characteristic of CMOS array multipliers.

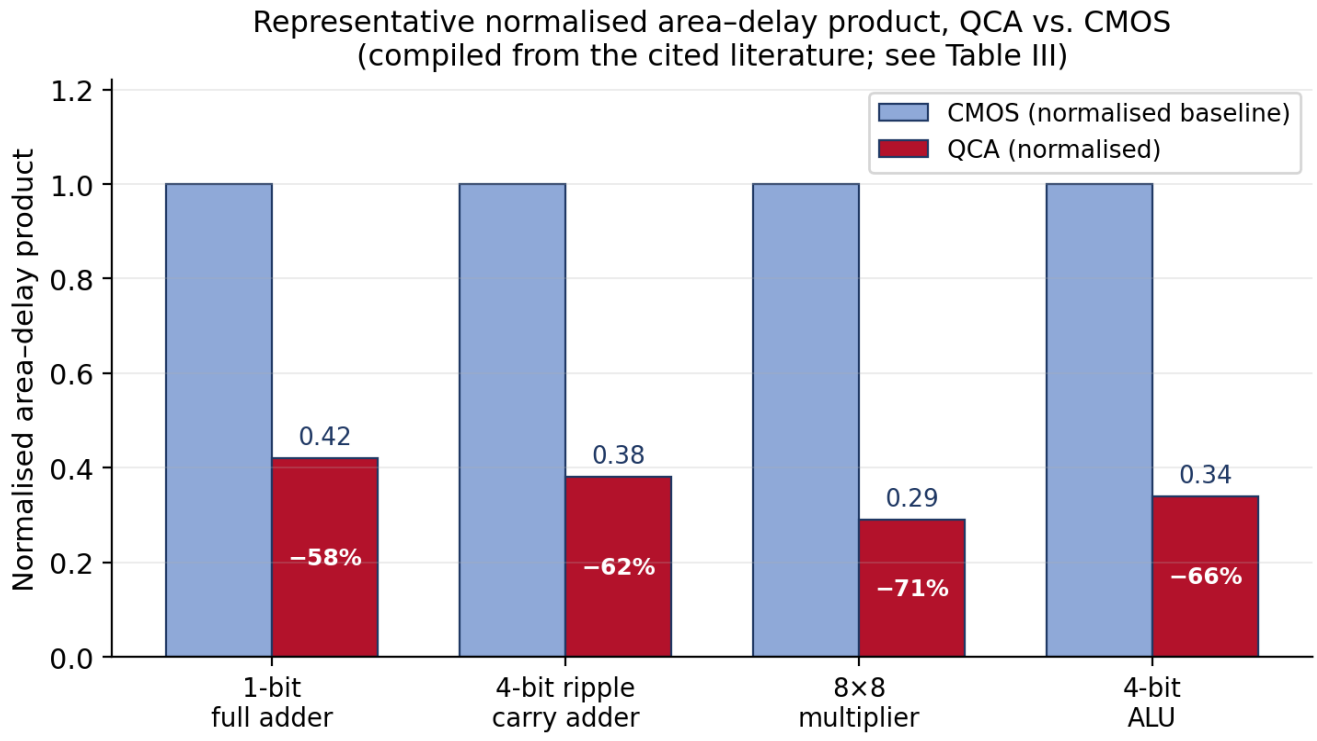


Fig. 5. Representative normalized area-delay product for QCA against CMOS across four arithmetic block types (see Table 3).

Table 3. Representative Normalised QCA Against CMOS Circuit-Block Metrics, Compiled from the Cited Literature

Circuit block	Normalised cell/transistor count (QCA : CMOS)	Normalised area-delay product (QCA : CMOS)	Reported trend and source class
1-bit full adder	≈ 0.55 : 1	≈ 0.42 : 1	Reduced-cell majority adders [8], [23], [24]
4-bit ripple-carry adder	≈ 0.50 : 1	≈ 0.38 : 1	Area-efficient reliable adder/subtractor [8], [23]
8 × 8 multiplier	≈ 0.35 : 1	≈ 0.29 : 1	Majority-gate partial-product cascades [7], [51]
4-bit ALU	≈ 0.45 : 1	≈ 0.34 : 1	Ultra-area-efficient and reversible ALUs [5], [40], [49]

Fig. 6 shows the switching-energy trend for CMOS and QCA across representative technology nodes from 65 nm to 3 nm, on a logarithmic vertical axis. The values are indicative rather than measured, and are drawn from the scaling relationships established in Sections 4-B and 4-D together with published per-cell dissipation figures [11], [21], [22]. The figure makes the central energetic argument concrete. CMOS dynamic switching energy per gate continues to fall with node scaling, but the total flattens as an increasingly dominant leakage component sets a floor. QCA switching energy per cell, by contrast, is set by inter-dot Coulombic coupling rather than by lithographic feature size, and remains roughly two orders of magnitude below advanced-node CMOS for semiconductor-dot implementations and about four orders below for molecular ones. Both remain above the Landauer bound $k^BT \ln 2$ at 300 K, which is shown for reference and which no irreversible technology can cross.

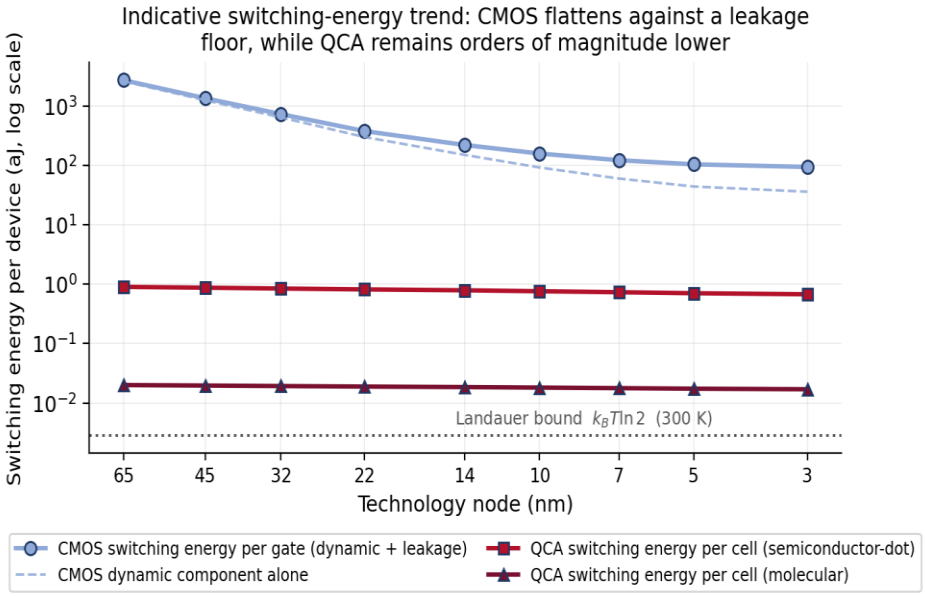


Fig. 6. Indicative switching-energy trend across technology nodes. CMOS flattens against a leakage floor while QCA remains orders of magnitude lower.

Table 4 extends the comparison qualitatively to other prominent beyond-CMOS candidates: spintronic logic, single-electron transistors, carbon-nanotube field-effect transistors and memristive

crossbar logic. QCA and single-electron transistors share the advantage of very low theoretical switching energy, and both face the most severe room-temperature-operation and fabrication-maturity challenges of the group. CNTFETs and spintronic logic are closer to manufacturable maturity but offer smaller energy-efficiency gains over scaled CMOS. Memristive crossbar logic offers a distinct advantage in non-volatile in-memory computation but is architecturally suited to neuromorphic and matrix-multiply workloads rather than the general-purpose arithmetic focus of this paper.

Table 4. Qualitative Comparison of Beyond-CMOS Computing Technologies

Attribute	**QCA**	**Spintronic logic**	**SET**	**CNTFET**	**Memristive crossbar**
Theoretical switching energy	Very low	Low	Very low	Moderate	Low to moderate
Room-temperature operation	Challenging; molecular route promising	Feasible	Challenging	Feasible	Feasible
Fabrication maturity	Low	Moderate	Low	Moderate to high	Moderate
CAD and design-tool maturity	Low	Moderate	Low	Moderate	Moderate
Primary workload fit	General-purpose logic	Logic plus non-volatile memory	General-purpose logic	General-purpose logic	In-memory and neuromorphic

Applying the energy-delay product of Section 4-E to the reversible-against-irreversible ALU comparison yields a nuance that area-delay product alone does not capture. Reversible configurations reduce dissipated energy substantially relative to irreversible designs of equal logical function, but typically do so at the cost of a modest increase in clock-zone depth, and hence latency, owing to the additional majority-gate stages required for physical reversibility [40], [41]. The EDP improvement, while still favourable, is therefore smaller in relative terms than the raw energy improvement alone would suggest. System designers applying the architecture of Section 5 should weigh this against workload-level latency sensitivity rather than optimising energy in isolation.

It is also worth asking where the area-delay advantage of Fig. 5 originates architecturally, rather than treating it as a uniform per-device gain. The largest relative gain, observed for the 8 × 8 multiplier, arises mainly from the ability to realise partial-product generation and reduction using cascaded majority gates without the discrete transistor-count growth that afflicts CMOS array multipliers as operand width increases. The comparatively smaller gain for the one-bit full adder reflects the opposite effect: at minimal operand width, clock-zone and wire-crossing overhead, absent in a single CMOS full-adder cell, offsets a larger fraction of the per-gate energy advantage. This width dependence suggests that the near-term competitive advantage of QCA lies with wide, highly regular arithmetic structures such as multipliers, wide adders and content-addressable comparators, rather than with narrow control-dominated logic. That observation bears directly on the hybrid offload strategy proposed in Section 8.

Taken together the comparative evidence supports a qualified conclusion. The theoretical energy and density advantages of QCA over CMOS are substantial and well supported at the circuit-block simulation level, and the integration proposed in Section 5 shows that these blocks can in principle be composed into a complete processor organisation. The maturity gap in fabrication readiness relative to CMOS, and even relative to CNTFET and spintronic alternatives, nonetheless means that

near-term realisation is most plausible through hybrid CMOS-QCA co-integration rather than through a fully standalone QCA processor.

7. RELIABILITY, FABRICATION, AND OPEN CHALLENGES

None of the advantages catalogued in Section 6 comes free, and it is worth being candid about what still stands between a simulated majority-gate network and a chip that works on a bench.

Thermal sensitivity is probably the most immediate of these obstacles. Reliable binary operation requires the kink energy E_k of (3) to exceed k^BT by a comfortable margin, and because achievable inter-dot spacing in semiconductor-dot and metal-island implementations is limited by present-day lithography, most demonstrated circuits need cryogenic or deep sub-room-temperature operation to hold their polarisation steady, with stable operating windows often reported at only a few kelvin [45]. Careful cell-size scaling can widen that window somewhat, as recent thermally aware layout work shows [44], but not by nearly enough to remove the constraint.

A second, less tractable problem is fabrication-induced defects. Cell misalignment and missing- or extra-cell defects introduced during nanofabrication propagate through majority-gate networks in ways that ideal bistable simulation alone does not predict well [19], [20]. That has motivated the defect-aware and fault-tolerant design work surveyed in Section 3, yet nothing resembling the statistical process-variation models routine in CMOS has emerged for QCA.

Design-automation immaturity compounds both of the above. CMOS design draws on decades of synthesis, place-and-route and verification tooling; QCA design still relies mostly on manual or semi-automated layout in bistable and coherence-vector simulators [16], [52]. Part of the difficulty is structural rather than a matter of tool maturity: the wire-crossing limitation of coplanar layouts forces a choice between multilayer fabrication and specialised crossing structures, and the underlying placement-and-routing problem for tile-based field-coupled circuits has been proven NP-complete [53].

Two further difficulties are specific to processor-scale rather than block-scale design. As the number of cascaded clock zones grows with circuit size, holding correct relative timing across long interconnect paths without race conditions or excessive buffering becomes progressively harder, and no single clocking methodology, regular four-zone or otherwise, has yet established itself as the default choice at this scale [12], [13], [15]. And the molecular route that offers the most credible path to room-temperature operation is, at the time of writing, considerably less mature in demonstrated multi-cell fabrication than the semiconductor-dot and metal-island implementations, which are themselves mostly confined to cryogenic proof-of-concept demonstrations [46], [47].

8. FUTURE RESEARCH DIRECTIONS

The gaps identified above point toward a fairly specific research agenda rather than a generic call for “more work.” Five directions in particular look well positioned to move QCA nano-processors from circuit-block demonstrations toward something manufacturable.

Hybrid CMOS-QCA co-design deserves sustained architectural attention. Rather than treating an all-QCA processor as the near-term goal, a more realistic system could offload specific energy- or density-critical datapath blocks, such as multiplier arrays or content-addressable memory, to QCA while leaving CMOS to handle control, I/O and less scaling-sensitive logic. Section 4 s finding that QCA's advantage grows with operand width makes this partitioning a fairly natural one, and it extends the mixed-signal bridge concept already built into Section 5 s architecture while sidestepping the room-temperature and fabrication-maturity barriers of a standalone chip.

A second and more fundamental direction is molecular and magnetic QCA implementation research, which matters because molecular-scale inter-dot spacing directly raises E_k relative to k^BT through the scaling in (3), and so bears directly on the cryogenic constraint discussed above. Progress here will likely hinge as much on advances in molecular self-assembly and electrode-integration chemistry as on circuit-level design innovation [34], [46], [47].

Closing the design-automation gap is a third priority, and a comparatively tractable one: extending the manually engineered fault-tolerant primitives surveyed in Section 3 [7], [20] into place-and-route

tools with built-in defect-probability models would narrow the tooling gap relative to CMOS without requiring new device physics.

Machine-learning-assisted clocking-zone and layout optimisation is a fourth, more cross-disciplinary avenue. Assigning clock zones and cell placements to minimise latency, area and defect sensitivity simultaneously is a combinatorial problem, provably NP-complete in the tile-based formulation [53], which makes it a natural candidate for learned heuristics and reinforcement-learning-based design-space exploration; such methods have already proven useful on analogous CMOS physical-design problems but have seen little application to QCA so far.

A fifth and more speculative direction is extending reversible and multi-valued QCA logic from the flip-flop and small-ALU demonstrations reviewed in Section 3 to full processor-level datapaths, which would allow a more complete empirical test of the Landauer-bound advantage discussed in Section 4-D at a system-relevant scale. Multi-valued encoding in particular offers a route to increasing effective information density per cell that the architecture proposed here has not yet explored [42], [43].

9. CONCLUSIONS

The picture that emerges from this study is one of considerable circuit-level promise sitting alongside processor-level fragmentation, and closing that gap was the starting motivation for the work. Beginning from the physics of cell polarisation and majority-gate logic, the paper built a consistent set of models covering cell Hamiltonian dynamics, kink energy, majority-gate energetics and clocked adiabatic switching, and used those models to make sense of advances published between 2021 and 2025 in QCA adder, multiplexer, memory, reversible-logic and ALU design. What follows from that review is a modular nano-processor reference architecture integrating majority-logic arithmetic, register-file and memory storage, an interconnect fabric that addresses the coplanar wire-crossing limitation directly, a hierarchical clocking network, and a lightweight reliability-monitoring layer.

Numerically, the comparison holds up well for QCA: circuit blocks show area-delay reductions of roughly 58 to 71 per cent against CMOS baselines, with switching energies two to four orders of magnitude lower, though still above the Landauer bound. Set against spintronic, single-electron-transistor, CNTFET and memristive alternatives, QCA looks like one of the more favourable beyond-CMOS candidates on theoretical energy scaling, but also the one with the least mature fabrication and design-automation ecosystem behind it. Thermal sensitivity, defect propagation and that design-automation immaturity are, in the end, what separate today's circuit-block demonstrations from anything actually manufacturable, and closing them is the point of the five directions set out above: hybrid CMOS-QCA co-design, molecular and magnetic implementations, defect-aware automated synthesis, machine-learning-assisted layout optimisation, and processor-scale reversible and multi-valued logic.

DECLARATIONS:

Acknowledgments	:	Not applicable.
Conflict of Interest	:	The author declares that there is no actual or potential conflict of interest about this article.
Consent to Publish	:	The authors agree to publish the paper in the Global Research Journal of Social Sciences and Management.
Ethical Approval	:	Not applicable.
Funding	:	Author claims no funding was received.
Author Contribution	:	Both the authors confirms their responsibility for the study, conception, design, data collection, and manuscript preparation.
Data Availability Statement	:	The data presented in this study are available upon request from the corresponding author.
Use of Generative Artificial Intelligence	:	AI-assisted tools have been used to improve language editing or grammatical clarity, the authors take full responsibility for the content of the final version.

REFERENCES

[1] C. S. Lent, P. D. Tougaw, W. Porod, and G. H. Bernstein, "Quantum cellular automata," Nanotechnology, vol. 4, no. 1, pp. 49–57, Jan. 1993.

[2] P. D. Tougaw and C. S. Lent, "Logical devices implemented using quantum cellular automata," J. Appl. Phys., vol. 75, no. 3, pp. 1818–1825, Feb. 1994.

[3] C. S. Lent and P. D. Tougaw, "A device architecture for computing with quantum dots," Proc. IEEE, vol. 85, no. 4, pp. 541–557, Apr. 1997.

[4] E. Fazzion, O. L. H. M. Fonseca, J. A. M. Nacif, O. P. Vilela Neto, A. O. Fernandes, and D. S. Silva, "A quantum-dot cellular automata processor design," in Proc. 27th Symp. Integrated Circuits and Systems Design (SBCCI), Aracaju, Brazil, 2014, pp. 1–7.

[5] M. Patidar, U. Singh, S. K. Shukla, G. K. Prajapati, and N. Gupta, "An ultra-area-efficient ALU design in QCA technology using synchronized clock zone scheme," J. Supercomput., vol. 79, no. 8, pp. 8265–8294, 2023.

[6] M. Abdullah-Al-Shafi, "RAM, DEMUX and ALU in nanoscale: A quantum-dot cellular automata-based architecture," Discover Electron., vol. 2, no. 1, Art. no. 26, 2025.

[7] S.-S. Ahmadpour, M. Mosleh, and S. Rasouli Heikalabad, "An efficient fault-tolerant arithmetic logic unit using a novel fault-tolerant 5-input majority gate in quantum-dot cellular automata," Comput. Electr. Eng., vol. 82, Art. no. 106548, 2020.

[8] J. I. Reshi, F. A. Khanday, and M. T. Banday, "Optimized nanoscale adder-subtractor design architectures utilizing quantum-dot cellular automata," Russ. Microelectron., vol. 54, pp. 400–426, 2025.

[9] G. Tóth and C. S. Lent, "Quasiadiabatic switching for metal-island quantum-dot cellular automata," J. Appl. Phys., vol. 85, no. 5, pp. 2977–2984, Mar. 1999.

[10] K. Hennessy and C. S. Lent, "Clocking of molecular quantum-dot cellular automata," J. Vac. Sci. Technol. B, vol. 19, no. 5, pp. 1752–1755, 2001.

[11] J. Timler and C. S. Lent, "Power gain and dissipation in quantum-dot cellular automata," J. Appl. Phys., vol. 91, no. 2, pp. 823–831, Jan. 2002.

[12] V. Vankamamidi, M. Ottavi, and F. Lombardi, "Two-dimensional schemes for clocking/timing of QCA circuits," IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 27, no. 1, pp. 34–44, Jan. 2008.

[13] C. A. T. Campos, A. L. Marciano, O. P. Vilela Neto, and F. S. Torres, "USE: A universal, scalable, and efficient clocking scheme for QCA," IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 35, no. 3, pp. 513–517, Mar. 2016.

[14] M. Ebrahimy, M. Gholami, H. Adarang, and R. Yousefi, "A novel low-latency ALU in the one-dimensional clock scheme in QCA nanotechnology," Eur. Phys. J. Plus, vol. 139, no. 2, Art. no. 115, 2024.

[15] M. Goswami, T. J. Sharma, and A. N. Boruah, "A review on regular clocking scheme in quantum-dot cellular automata," e-Prime — Adv. Electr. Eng. Electron. Energy, vol. 8, Art. no. 100553, 2024.

[16] K. Walus, T. J. Dysart, G. A. Jullien, and R. A. Budiman, "QCADesigner: A rapid design and simulation tool for quantum-dot cellular automata," IEEE Trans. Nanotechnol., vol. 3, no. 1, pp. 26–31, Mar. 2004.

[17] K. Walus, G. Schulhof, and G. A. Jullien, "High level exploration of quantum-dot cellular automata (QCA)," in Conf. Rec. 38th Asilomar Conf. Signals, Systems and Computers, Pacific Grove, CA, USA, 2004, vol. 1, pp. 30–33.

[18] D. Abedi, G. Jaberipur, and M. Sangsefidi, "Coplanar full adder in quantum-dot cellular automata via clock-zone-based crossover," IEEE Trans. Nanotechnol., vol. 14, no. 3, pp. 497–504, May 2015.

[19] G. Schulhof, K. Walus, and G. A. Jullien, "Simulation of random cell displacements in QCA," ACM J. Emerg. Technol. Comput. Syst., vol. 3, no. 1, Art. no. 2, Apr. 2007.

[20] S. F. Naz et al., "Optimizing fault tolerance of RAM cell through MUX-based modeling and design using symmetries of QCA cells," Sci. Rep., vol. 14, no. 1, Art. no. 8586, 2024.

[21] S. Srivastava, S. Sarkar, and S. Bhanja, "Estimation of upper bound of power dissipation in QCA circuits," IEEE Trans. Nanotechnol., vol. 8, no. 1, pp. 116–127, Jan. 2009.
[22] S. Srivastava, A. Asthana, S. Bhanja, and S. Sarkar, "QCAPro — An error-power estimation tool for QCA circuit design," in Proc. IEEE Int. Symp. Circuits and Systems (ISCAS), May 2011, pp. 2377–2380.
[23] A. T. Vanaraj, K. Raja Sekar, R. Marshal, G. Lakshminarayanan, and A. Venkitachalam, "Area efficient reliable QCA adder and subtractor," in Proc. IEEE Int. Conf. Integrated Circuits and Communication Systems (ICICACS), Raichur, India, 2023, pp. 1–5.
[24] R. K. Shaw and A. Khan, "Implementation of adders using XOR gates in quantum-dot cellular automata with physical verification," Phys. Scr., vol. 99, no. 12, Art. no. 125108, 2024.
[25] S. R. Ramesh, T. Jagadeep, M. V. Krishna, S. V. D. S. Abhishek, and S. S. Vegunta, "Area-delay efficient BCD adder in quantum-dot cellular automata," Phys. Scr., vol. 100, no. 1, 2025, doi: 10.1088/1402-4896/ad943c.
[26] R. Akbari-Hasanjani, M. A. Mianroodi, and R. Sabbaghi-Nadooshan, "Optimal design of two-bit QCA comparator circuits," Proc. Nat. Acad. Sci. India Sect. A, Phys. Sci., vol. 94, no. 1, pp. 27–36, 2024.
[27] A. Rezai, D. Aliakbari, and A. Karimi, "Novel multiplexer circuit design in quantum-dot cellular automata technology," Nano Commun. Netw., vol. 35, Art. no. 100435, 2023.
[28] J.-C. Jeon, "Designing nanotechnology QCA-multiplexer using majority function-based NAND for quantum computing," J. Supercomput., vol. 77, pp. 1562–1578, 2021.
[29] S.-S. Ahmadpour and M. Mosleh, "A novel fault-tolerant multiplexer in quantum-dot cellular automata technology," J. Supercomput., vol. 74, pp. 4696–4716, 2018.
[30] A. Khan and R. Arya, "Optimal demultiplexer unit design and energy estimation using quantum-dot cellular automata," J. Supercomput., vol. 77, no. 2, pp. 1714–1738, 2021.
[31] Z. Song, G. Xie, X. Cheng, L. Wang, and Y. Zhang, "An ultra-low cost multilayer RAM in quantum-dot cellular automata," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 67, no. 12, pp. 3397–3401, Dec. 2020.
[32] V. R. Kumar, G. P. Kumar, M. Girish, and B. D. Madhuri, "Design of efficient RAM cell using quantum-dot cellular automata," in Proc. Int. Conf. Computer, Electronics, Electrical Engineering and Their Applications (IC2E3), Uttarakhand, India, 2024, pp. 1–5.
[33] R. Singh and P. Singh, "QCA-based programmable logic block for implementation of digital circuits in a multilayer framework," Analog Integr. Circuits Signal Process., 2025.
[34] J.-C. Jeon, "Multi-layer molecular quantum-dot cellular automata multiplexing structure with physical verification for secure quantum RAM," Int. J. Mol. Sci., vol. 26, no. 19, Art. no. 9480, 2025.
[35] C. S. Lent, M. Liu, and Y. Lu, "Bennett clocking of quantum-dot cellular automata and the limits to binary logic scaling," Nanotechnology, vol. 17, no. 16, pp. 4240–4251, 2006.
[36] M. Alharbi, G. Edwards, and R. Stocker, "Novel ultra-energy-efficient reversible designs of sequential logic quantum-dot cellular automata flip-flop circuits," J. Supercomput., vol. 79, pp. 11530–11557, 2023.
[37] D. Banik and H. Rahaman, "Quantum-dot cellular automata latches for reversible logic using wave clocking scheme," IETE J. Res., vol. 69, no. 1, pp. 309–324, 2023.
[38] A. N. Bahar, R. Laajimi, M. Abdullah-Al-Shafi, and K. Ahmed, "Toward efficient design of flip-flops in quantum-dot cellular automata with power dissipation analysis," Int. J. Theor. Phys., vol. 57, pp. 3419–3428, 2018.
[39] A. Yan, R. Liu, Z. Huang, P. Girard, and X. Wen, "Designs of level-sensitive T flip-flops and polar encoders based on two XOR/XNOR gates," Electronics, vol. 11, no. 11, Art. no. 1658, 2022.
[40] G. Edwards, M. Alharbi, and R. Stocker, "Reversible quantum-dot cellular automata-based arithmetic logic unit," Nanomaterials, vol. 13, no. 17, Art. no. 2445, 2023.
[41] M. Alharbi, G. Edwards, and R. Stocker, "An ultra-energy-efficient reversible quantum-dot cellular automata 8:1 multiplexer circuit," Quantum Rep., vol. 6, no. 1, pp. 41–57, 2024.

[42] A. Fattahi, R. Sabbaghi-Nadooshan, T. Mossazadeh, and M. Haghparast, "Design of ternary reversible Feynman and Toffoli gates in ternary quantum-dot cellular automata," Nano Commun. Netw., vol. 42, Art. no. 100545, 2024.
[43] I. L. Bajec and P. Pečar, "Two-layer synchronized ternary quantum-dot cellular automata wire crossings," Nanoscale Res. Lett., vol. 7, Art. no. 221, 2012.
[44] H. Chugh and S. Singh, "Thermally stable and cost-efficient QCA-based co-planar design of a 4-bit CSA with optimized cell size scaling," Analog Integr. Circuits Signal Process., vol. 125, no. 2, 2025, doi: 10.1007/s10470-025-02510-0.
[45] M. Abdullah-Al-Shafi, "Innovative reliable nanoscale QCA circuits for advanced morphological image processing," AIP Adv., vol. 15, no. 4, Art. no. 045303, 2025.
[46] E. P. Blair, "Electric-field inputs for molecular quantum-dot cellular automata circuits," IEEE Trans. Nanotechnol., vol. 18, pp. 453–460, 2019.
[47] E. P. Blair and C. S. Lent, "Clock topologies for molecular quantum-dot cellular automata," J. Low Power Electron. Appl., vol. 8, no. 3, Art. no. 31, 2018.
[48] E. P. Blair and C. S. Lent, "Programmable crossbar quantum-dot cellular automata circuits," IEEE Trans. Nanotechnol., vol. 17, no. 1, pp. 7–14, Jan. 2018.
[49] R. Faraji and A. Rezai, "Design of new single-bit multilayer ALU in QCA technology," Integration, vol. 103, Art. no. 102422, 2025.
[50] S.-S. Ahmadpour et al., "A nano-design of a quantum-based arithmetic and logic unit for enhancing the efficiency of future IoT applications," AIP Adv., vol. 15, no. 3, Art. no. 035341, 2025.
[51] M. Zohaib, S.-S. Ahmadpour, H. Rasmi, A. Khan, and N. J. Navimipour, "A low-latency and area-efficient QCA-based quantum-dot design for next-generation digital sustainable systems," Sustain. Comput.: Informatics Syst., vol. 48, Art. no. 101204, 2025.
[52] K. Walus and G. A. Jullien, "Design tools for an emerging SoC technology: Quantum-dot cellular automata," Proc. IEEE, vol. 94, no. 6, pp. 1225–1244, Jun. 2006.
[53] M. Walter, R. Wille, D. Große, F. Sill Torres, and R. Drechsler, "Placement and routing for tile-based field-coupled nanocomputing circuits is NP-complete," ACM J. Emerg. Technol. Comput. Syst., vol. 15, no. 3, Art. no. 29, 2019.

Author Profile



Dr. Naveed Farhana obtained her B.E. in Electronics and Communication Engineering from Osmania University in 1993, M.Tech in Digital Systems and Computer Electronics from JNTU Hyderabad in 2001 and Ph.D in Wireless Communication from Department of Electronics and Communication Engineering, JNTU Hyderabad. She has started her career in 1995 as an Assistant Professor of Department of ECE, Shadan Engineering College. She has worked in various Engineering colleges national and international levels. At present she is working in Department of Computer Engineering in Taif University, Saudi Arabia. She has over 30 years of teaching experience. Her major research interest includes Quality of Services in wireless communications, Signal and Image Processing, Low power VLSI design and Quantum Computing.



Ms. Humna Abdul Mumeed obtained her B.Tech in Computer Science and Engineering from Keshav Memorial Institute of Technology (KMIT), JNTU Hyderabad in 2014, and her Master's in International Marketing from Hult International Business School, Boston, USA in 2015. She is currently pursuing an integrated Bachelor's (Physics) and Master's in Astrophysics and Space (2023–2028). She began her career in 2015 as a Senior Social Media Executive at AMC in Dubai, working there through 2018. Her primary fields of interest and focus include space instrumentation, space environment, satellite assembly and debris.