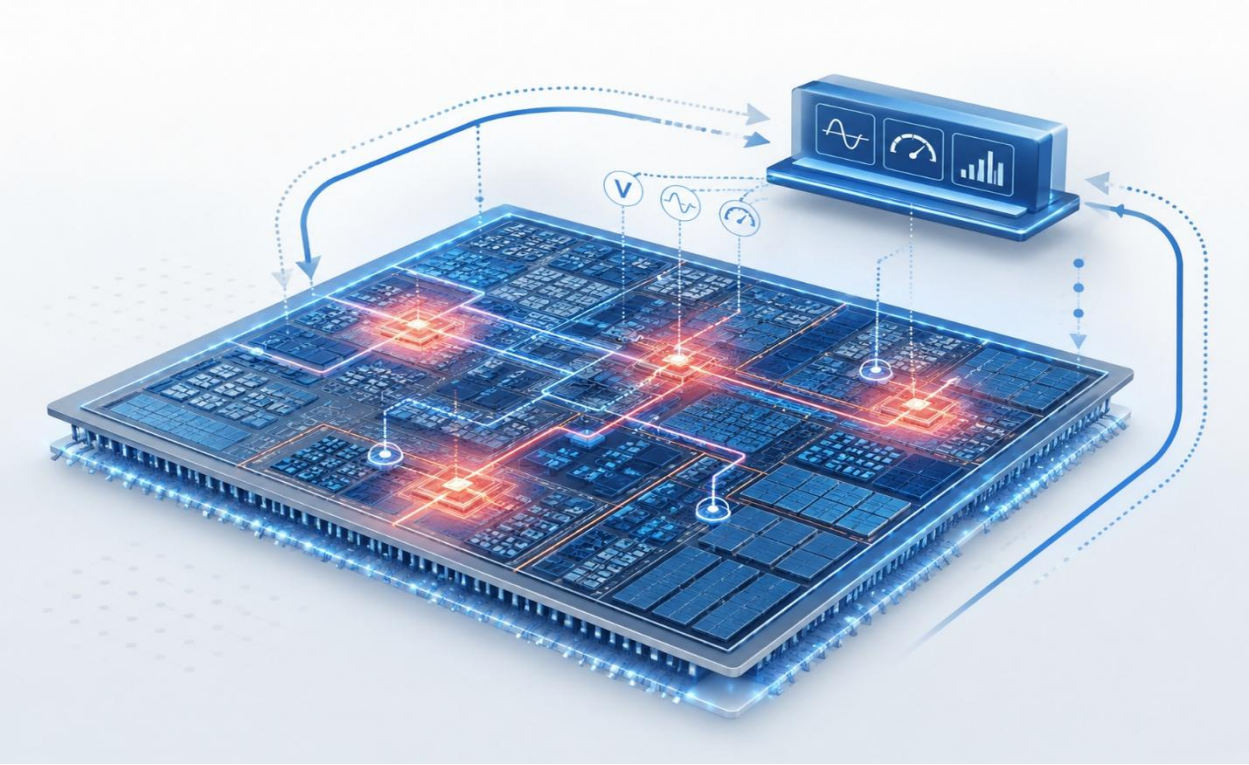
VegaSemi
Smart Silicon



Unified Silicon Runtime™ Power Architecture
From Static OPP to Runtime Control

Technical Foundations and Deployment Path for VegaPower™

White Paper | April 2026

Ben Zaryouh
Founder & CEO
VegaSemi
ben@vegasemi.com

Table of Contents

1 Executive Summary

Power management in modern System-on-Chip platforms is becoming fundamentally more difficult as semiconductor technology continues to scale. Advanced nodes, including FinFET and GAAFET generations, introduce increasingly complex behavior through device variability, process gradients, thermal stress, slower heat dissipation, aging effects, and growing interconnect sensitivity. These effects make timing and voltage margins more dynamic, localized, and difficult to manage using traditional assumptions.

At the same time, modern applications such as AI and other compute-intensive workloads impose highly variable and bursty activity patterns that further amplify these technology-node challenges. Rapid workload transitions, localized switching, and uneven thermal buildup create operating conditions that are increasingly difficult to capture through static characterization or indirect runtime approximation.

Conventional power-management methods, including static operating-point frameworks and proxy-based runtime techniques, provide useful control but remain fundamentally limited. Because they do not directly observe true functional timing behavior in the live execution fabric, they must preserve conservative guardbands to remain safe across uncertain runtime conditions.

That conservatism carries a broad system cost. It increases power consumption, drives area and infrastructure overdesign, reduces effective silicon utilization and yield opportunity, and can negatively affect thermal behavior and product lifespan.

This paper argues that these challenges require a different architectural approach. Rather than managing power primarily through pre-characterized assumptions or indirect proxies, modern SoCs increasingly require runtime control based on direct in-situ observation of real behavior in the functional fabric. VegaPower™ is introduced as a runtime power architecture built on that principle, enabling a shift from estimation-driven power management to execution-coupled, closed-loop control.

2 Introduction

Power management has become one of the defining constraints in modern System-on-Chip design. In earlier technology generations, it could often be treated primarily as an optimization problem: reduce energy where possible while preserving sufficient margin for safe operation. That assumption is becoming increasingly difficult to sustain. In modern SoCs, power management now sits at the intersection of performance, timing integrity, thermal behavior, reliability, and product economics.

This shift is being driven by two converging forces. The first is semiconductor scaling itself. As technologies continue through advanced FinFET and GAAFET generations, usable timing and voltage margin become increasingly sensitive to non-uniform device variation, thermal gradients, aging, interconnect effects, and localized operating conditions. The second is workload behavior. AI-driven and other compute-intensive applications create highly variable and bursty activity patterns that stress the silicon in ways that are increasingly difficult to capture through static assumptions alone.

Under these conditions, conventional approaches to power management face growing tension between safety and efficiency. Broad guardbands remain necessary where runtime behavior cannot be observed directly, but those guardbands carry an increasing cost across power, thermal design, silicon utilization, and product lifetime. As a result, the question is no longer only how to tune voltage and frequency more effectively, but how to build a runtime control architecture that remains meaningfully aligned with real silicon behavior.

This paper examines that problem and develops the case for a different runtime power-management model. It first reviews the limitations of conventional approaches, then introduces VegaPower™ as an architectural framework built around direct in-situ observability, closed-loop runtime control, and lifecycle-aware adaptation. The goal is not simply to improve an existing DVFS policy, but to outline a more scalable foundation for power management in advanced-node SoCs.

3 Limitations of Conventional SoC Power Management

Key Observations:

- Advanced-node silicon exhibits increasingly dynamic timing behavior due to non-uniform variation, thermal confinement, slow heat diffusion, localized hotspot behavior, aging, interconnect effects, and electromigration.
- Modern workloads intensify these effects through bursty, localized, and highly variable execution behavior.
- OPP-based control remains widely deployed, but its pre-characterized nature forces conservative guardbanding.
- OPP aided by runtime telemetry improves operating-point selection, but because it still lacks direct visibility into true timing margin, it remains structurally inefficient and continues to carry conservative guardbands.
- Existing runtime techniques improve adaptation, yet remain limited by proxy behavior, correlation drift, probabilistic inference, or recovery-oriented operation.
- The missing capability is a scalable architecture for direct, execution-coupled runtime margin observability and control.

3.1 Technology Context and Key Challenges

Modern System-on-Chip (SoC) power management now operates in a technology regime that is fundamentally more demanding than in prior generations. At modern advanced nodes, including 5nm, 3nm, and beyond, silicon behavior is increasingly affected by pronounced non-uniform transistor variation across the die. Process gradients are no longer secondary effects; they directly influence local device behavior, timing distribution, and usable voltage margin. Variations in threshold voltage (Vth), carrier mobility, and related device characteristics create spatially distributed timing behavior that cannot be fully captured by broad design corners alone.

This challenge is compounded by thermal behavior, which becomes tightly intertwined with process variation from the outset. Thermal gradients emerge early and evolve continuously during operation, driven by localized activity, heterogeneous compute loading, bursty workloads, and hotspot formation. Because temperature directly impacts transistor speed, leakage, and reliability, these gradients create timing behavior that is strongly context-dependent and highly dynamic. In practice, process variation and thermal variation do not act independently; they interact and amplify timing uncertainty across the silicon fabric.

Aging mechanisms add a longer-term dimension to this instability. Bias Temperature Instability (BTI) and Hot Carrier Injection (HCI) progressively alter transistor characteristics over product lifetime, with degradation rates depending on local voltage stress, switching activity, duty cycle, and thermal history. As a result, path criticality can reorder over time, making long-term voltage-frequency management increasingly difficult to handle through static assumptions or fixed correlation models.

Interconnect effects have also become a first-order challenge in advanced technologies. Interconnect delay, RC effects, coupling, and localized voltage drop contribute materially to path timing uncertainty, and in many cases wire behavior becomes as important as transistor delay itself. This is particularly severe under dynamic workloads, where simultaneous switching and localized current demand can trigger rapid droops and timing stress. In addition, electromigration has become a more serious reliability and design constraint at modern nodes, further complicating long-term margin management and power-delivery integrity.

These technology-node effects are further intensified by the changing nature of modern applications. AI workloads, heterogeneous compute, bursty accelerators, and rapidly shifting execution phases create highly variable switching behavior across space and time. Unlike more predictable legacy workloads, modern applications can produce abrupt activity transitions, localized current surges, uneven thermal buildup, and path sensitization patterns that are difficult to anticipate through static characterization alone. In this sense, application demand does not merely sit on top of technology scaling challenges; it actively worsens them by exciting the silicon in more dynamic and less predictable ways.

Taken together, these effects create a central problem for modern power management: true timing margin is no longer static, uniform, or easily inferred. It evolves across space, time, workload, environmental conditions, and product lifetime. Any power-management approach that does not directly track this runtime reality must compensate through conservative margins, indirect estimation, or reduced control ambition. That is the fundamental limitation of many conventional approaches.

3.2 Pre-Characterized Operating Frameworks

Pre-characterized operating frameworks remain the dominant foundation of practical SoC power management. In these approaches, voltage and frequency behavior is largely determined from design-time analysis, silicon characterization, and predefined operating limits rather than direct observation of true functional timing margin during runtime. The most common realization of this model is OPP or static table-based power management.

3.2.1 OPP / Static Table-Based Power Management

Operating Performance Point (OPP) or static-table-based power management remains the most common industry deployment model because it is simple, familiar, institutionally accepted through established signoff and validation practices, and relatively easy to qualify for product deployment. In this approach, voltage-frequency pairs are established primarily through design-time analysis, characterization, silicon bring-up, and production screening, then used as approved operating states during field operation. Its practicality and maturity make it attractive, but those same characteristics also define its limits.

Key limitations include:

- **Design-time derivation:** Safe operating points are established largely before

deployment and therefore cannot reflect the continuously changing runtime state of the silicon.
- **Worst-case orientation:** OPP tables must remain safe across process spread, temperature variation, voltage noise, workload diversity, transient events, and lifetime degradation.
- **Pessimistic margin stacking:** Multiple guardband assumptions are combined conservatively even when they rarely coincide in real operation.
- **No direct runtime timing visibility:** Safety is inferred from pre-approved voltage-frequency relationships rather than direct observation of actual timing margin during execution.
- **Weak workload specificity:** Different applications sensitize different structures, yet a static OPP is assumed broadly valid across many execution conditions.
- **Poor localized adaptation:** Coarse operating points cannot reflect regional timing stress, local droops, or hotspot-driven variation across the die.
- **Weak lifetime adaptation:** BTI, HCI, and other aging effects are covered mainly through added pessimism or occasional coarse recalibration.
- **Early-life overmargining:** Silicon often carries end-of-life margin from the beginning of product life, wasting available headroom during early operation.
- **Systematic power inefficiency:** Excess voltage increases both dynamic power and leakage, often reinforcing thermal self-amplification.
- **Infrastructure overdesign:** PDN, decoupling, regulator response, packaging, and thermal solutions must all be sized for inflated worst-case assumptions.
- **Poor scalability:** As advanced nodes, workload diversity, and lifetime sensitivity increase, static tables become increasingly blunt and inefficient.
- **Coarse safety-envelope behavior:** OPP remains useful as a baseline deployment model, but it is fundamentally limited in how closely it can approach true silicon-optimal operation.

3.2.2 OPP Aided by Runtime Information

Many modern SoCs do not rely on static OPP alone. Instead, runtime governors refine operating-point selection using temperature, utilization, activity counters, workload indicators, or other telemetry. This improves adaptability relative to purely static operation, but the underlying control model remains fundamentally pre-characterized: runtime logic selects among pre-approved states rather than directly observing and regulating true functional timing margin.

Key limitations include:

- **Pre-bounded operating envelope:** Runtime refinement still operates within a design-time-approved voltage-frequency space.
- **Indirect telemetry dependence:** Temperature, utilization, counters, and workload indicators may correlate with timing stress, but they are not direct timing truth.
- **No direct visibility into functional margin:** The controller still does not

know the actual timing distance to failure under live execution.
- **Policy-limited adaptation:** Governor behavior is constrained by predefined state transitions, heuristics, and safety thresholds.
- **Weak sensitivity to path sensitization:** Real timing stress depends on specific execution conditions that generic runtime metrics may not capture.
- **Limited handling of localized behavior:** Runtime telemetry is often too coarse to capture local droops, localized thermal buildup, hotspot persistence, or path-specific stress.
- **Residual dependence on conservative OPP tables:** Even with runtime refinement, selected states still inherit the pessimism embedded in the original characterized envelope.
- **Weak lifetime adaptation:** Aging effects are still handled conservatively unless significant recalibration infrastructure is added.
- **Control lag under fast events:** Runtime governors are generally not designed for immediate response to rapid droops or abrupt workload transitions.
- **Growing policy complexity:** Adding more telemetry and decision rules increases software and tuning burden without eliminating the core observability gap.
- **Still estimation-driven:** The system becomes more adaptive, but it remains based on indirect interpretation rather than direct execution-coupled timing knowledge.

3.3 Existing Runtime Techniques

Existing runtime techniques attempt to improve power-management adaptation by incorporating runtime sensing, correlation, inference, or in-situ timing awareness. While these approaches move beyond static OPP-based control, they still remain limited by indirect observability, correlation fragility, recovery-oriented behavior, or incomplete visibility into true functional timing margin.

3.3.1 Ring Oscillator (RO)-Based AVFS

RO-based AVFS methods are attractive because they are relatively simple and provide some runtime indication of silicon speed. However, they remain fundamentally proxy-based and therefore structurally limited.

Key limitations include:

- **Indirect proxy behavior:** RO delay does not directly represent real functional critical paths or true execution conditions.
- **Structural mismatch to real paths:** Even when RO-like monitors are built using more representative gate structures rather than simple inverter chains, they still cannot faithfully capture the logic

depth, loading, routing, interconnect, coupling, and path diversity of the full population of real critical paths in a modern SoC.
- **Limited spatial representativeness:** Fixed RO placement cannot capture the distributed timing behavior of a large modern SoC.
- **Hotspot blindness:** Sparse monitors can miss localized thermal hotspots, local droops, and workload-specific stress conditions.
- **Decoupled from real execution:** ROs do not capture data-dependent switching, actual path sensitization, or the temporal patterns that determine which paths are truly stressed.
- **Weak coverage of rare events:** Important timing failures often arise under specific path sensitization conditions that a generic oscillator cannot reproduce reliably.
- **Reliance on imperfect correlation models:** Safe voltage-frequency decisions still depend on translation models that must absorb uncertainty from variation, temperature, aging, noise, and functional mismatch.
- **Measurement ambiguity:** RO behavior is itself sensitive to supply noise, jitter, and local environmental effects.
- **Limited utility under fast transients:** Meaningful RO measurement often requires observation windows, averaging, or filtering, reducing responsiveness when rapid action is needed.
- **Slow control loop:** Sensing, filtering, decision, and regulator response are slow relative to fast droop events.
- **Heavy characterization burden:** Significant correlation work is required across PVT, workloads, noise, and aging.
- **Lifecycle fragility:** ROs and real functional paths do not age identically, so correlation can drift over time.
- **Persistent guardbanding:** Because uncertainty remains, RO-based AVFS still requires conservative margin and cannot safely collapse guardbands to the level needed for true fine-grain runtime optimization.

3.3.2 Replica / Critical Path Replica-Based AVFS

Replica-path techniques attempt to improve on generic proxies by tracking structures intended to resemble real timing-critical behavior more closely. Even so, they remain dependent on correlation assumptions that are difficult to sustain across workloads and lifetime.

Key limitations include:

- **Replica-path mismatch under real operation:** Even a closely matched replica path can diverge from the original functional path because workload-dependent activation, selfheal-ting, local operating conditions, and aging do not evolve identically.
- **Limited early-detection fidelity:** Added skew or replica pessimism cannot guarantee that the replica will always indicate insufficient margin before the true functional path becomes critical, and in practice the information provided is often coarse rather than a precise measure of actual functional timing margin.

- **Residual guardband requirement:** Because mismatch cannot be eliminated with certainty, conservative voltage-frequency margin must still be retained.

3.3.3 AI-Based AVFS

AI-based AVFS attempts to predict safe voltage-frequency settings using telemetry, counters, activity information, or learned workload behavior. While this can increase flexibility, it remains inference-based rather than tied directly to real-time timing truth.

Key limitations include:

- **Inference from indirect observables:** AI-based AVFS does not directly measure actual functional timing margin; instead, it infers timing stress from proxies such as performance counters, thermal sensors, and activity monitors, which are not equivalent to direct timing truth.
- **Workload non-stationarity:** Real SoC operation evolves over time, and field workloads often differ from characterization assumptions.
- **Difficulty capturing rare corner cases:** Important failures may be driven by uncommon combinations of data, activity, localized thermal behavior, supply noise, and path sensitization.
- **Feature-selection blind spots:** Critical determinants of timing stress may be hidden, weakly observable, or omitted from the model.
- **Generalization uncertainty:** Good performance on known workloads does not guarantee robustness on unseen instruction mixes or runtime conditions.
- **Model drift:** Aging, software evolution, process spread, and deployment diversity can degrade model accuracy over time.
- **High validation complexity:** Proving safe behavior across relevant workloads, PVT conditions, and lifetime scenarios is difficult.
- **Inference latency:** AI can help slower adaptation, but it is poorly suited to very fast droops or abrupt timing stress.
- **Infrastructure overhead:** Additional hardware, firmware, memory, telemetry pipelines, and software support add area, power, and verification cost.
- **Training and maintenance burden:** Robust deployment requires training, recalibration, retraining, and fallback management.
- **Probabilistic control decisions:** Because inference is not deterministic, conservative fallback mechanisms and retained guardbands remain necessary in practice.

3.3.4 In-Situ Timing Techniques (e.g., Razor-Class Approaches)

In-situ timing techniques move closer to true runtime timing awareness than static OPP, telemetry-guided governors, or indirect proxies. Razor-class approaches, for example, detect timing errors or near-fail behavior at execution time and use that information to guide voltage

reduction. This is an important step toward direct runtime timing awareness, but it introduces a different set of limitations that constrain broad deployment as a general SoC power architecture.

Key limitations include:

- **Failure-and-recovery operation:** Razor-class techniques rely on detection of actual timing errors followed by replay, flush, rollback, or correction mechanisms.
- **Detection after critical stress:** The mechanism often reacts at or near the point of violation rather than maintaining proactive control of true timing margin.
- **Recovery overhead:** Replay, flush, rollback, or correction mechanisms can impose performance and energy penalties.
- **Architectural complexity:** Integration into high-performance pipelines, speculative machines, and heterogeneous SoCs can be difficult.
- **Clocking complexity:** Multi-sampling flops, shadow latches, and timing-error capture introduce non-trivial implementation and verification burden.
- **Limited coverage:** Not all timing-critical structures or execution contexts are easily instrumented.
- **Sensitivity to noise and metastability:** Distinguishing true timing events from noise-induced artifacts can be challenging.
- **Scalability constraints:** Broad application across large SoCs can create significant design and validation overhead.
- **Reactive rather than margin-aware:** These methods reveal that timing has been stressed, but they do not inherently provide a clean, continuous measure of functional timing margin.
- **Not ideal for general runtime power architecture:** While valuable in certain contexts, they are difficult to use as a universal, low-overhead, execution-wide runtime power-control framework.

3.3.5 Transition to a New Runtime Power-Control Paradigm

These approaches reveal a consistent pattern. OPP remains practical but fundamentally conservative. OPP aided by runtime information improves adaptation, yet remains bounded by indirect signals and pre-approved states. RO, replica, and AI-based AVFS add runtime sophistication, but remain proxy-based, correlation-dependent, or inference-driven. Razor-class in-situ techniques move closer to direct timing awareness, but remain recovery-oriented and difficult to scale as a general SoC runtime power architecture.

The central limitation is therefore not simply the lack of smarter policy. It is that existing approaches cannot scale effectively with the realities of modern technology nodes and highly dynamic AI-era workloads. As advanced FinFET and GAAFET generations intensify non-uniform variation, thermal confinement, slow heat diffusion, aging, interconnect sensitivity, and localized stress, indirect and pre-characterized techniques become increasingly misaligned with true runtime silicon behavior. Under the same conditions, bursty and heterogeneous workloads amplify path-specific timing stress in ways that coarse guardbands, indirect correlation, probabilistic inference, and recovery-after-failure schemes cannot manage efficiently or robustly.

What is missing is a scalable architecture that can directly observe and regulate true functional timing margin during real execution. That gap motivates the runtime power-control paradigm introduced in the next section.

4 VegaPower™ Architectural Paradigm

Key Observations:

- VegaPower introduces a shift from estimation-driven power management to execution-coupled runtime control based **on direct in-situ** observation.
- Functional-path observability replaces indirect proxies, enabling more accurate and responsive timing-margin awareness.
- Closed-loop runtime control allows continuous adaptation to workload, thermal behavior, and aging rather than reliance on pre-characterized operating points.
- Integration of fast transient mechanisms, including droop detection and correction (DDC), reduces dependence on static guardbands for supply integrity.
- Architecture supports lifecycle-aware adaptation through flexible critical-path selection and evolving observability coverage.

The previous section showed that conventional power-management approaches remain limited by pre-characterized assumptions, indirect observability, and residual uncertainty. As technology nodes continue to scale and workloads become more dynamic, these limitations become increasingly difficult to manage through incremental refinement alone. This section introduces VegaPower as a different architectural response: a runtime power-control paradigm built around direct observability of real behavior in the functional fabric and closed-loop adaptation during execution.

4.1 Architectural Shift: From Estimation to Runtime Observation

The limitations of conventional power-management approaches become more pronounced as SoC technology advances into modern FinFET and GAAFET nodes. Static operating frameworks such as OPP and fixed voltage-frequency tables remain widely deployed because they are practical and familiar, while proxy-based runtime methods attempt to improve adaptability through indirect sensing and correlation. However, both classes of approach become increasingly inadequate as process variability, localized gradients, aging behavior, interconnect sensitivity, and thermal complexity intensify at advanced nodes.

Thermal behavior is especially important in this transition. In modern SoCs, heat generation is increasingly localized, thermal confinement becomes more severe, and temperature escape is slower and less uniform across the silicon fabric. As a result, thermal gradients can persist and evolve in ways that are highly dependent on local activity, workload phase, and physical layout. This creates a runtime environment in which hotspots and timing stress develop in a distributed and time-varying manner that is difficult for indirect monitors to represent faithfully.

These conditions directly weaken the effectiveness of conventional control techniques. Proxy-based runtime methods, including oscillators, replicas, and telemetry-driven inference, remain limited because they do not directly observe the true timing behavior of the functional execution

fabric under these increasingly localized and thermally constrained conditions. Static OPP frameworks face a related challenge. Although they are often aided by runtime policy managers and temperature sensing, thermal propagation itself is relatively slow, and temperature sensors inevitably report an already-developed condition rather than the instantaneous state of the most timing-sensitive functional paths. In other words, the thermal information available to OPP-based control is inherently lagging relative to the fast and localized timing stress that must ultimately be managed.

For this reason, the problem is no longer simply one of refining existing governors or improving correlation models. Modern power management increasingly requires a different architectural foundation: one based on direct runtime observation of real behavior in the functional fabric itself. VegaPower is built on that principle. Rather than relying primarily on pre-characterized assumptions, indirect proxies, or lagging thermal indicators, it introduces a runtime control model in which real silicon behavior, real timing margin, and control response are directly linked during execution.

4.2 Core Architectural Principles

VegaPower is built around a set of architectural principles that distinguish it from static operating frameworks, proxy-based runtime methods, and recovery-oriented in-situ timing techniques. At the foundation of the architecture is **direct in-situ observability within the functional execution fabric**, which allows power control to be based on real silicon behavior rather than indirect estimation.

- **In-situ, execution-coupled observability:** Margin-relevant behavior is observed directly within the functional fabric during real execution rather than inferred from proxies, external telemetry, or offline models.
- **Closed-loop runtime control:** Regulation is continuous and tied to live silicon conditions rather than open-loop tables or semi-closed supervisory control.
- **Fine-grain adaptation:** Voltage and frequency track continuous margin evolution rather than moving only among coarse pre-approved states.
- **Execution-relevant response time:** The architecture supports response at time scales relevant to real silicon stress events, including fast transient droop conditions, rather than relying only on delayed thermal or averaged telemetry loops.
- **Integrated transient integrity management:** Fast events such as supply droops are handled as part of the runtime power architecture itself, including mechanisms such as droop detection and correction (DDC), rather than being addressed only through static voltage margin.
- **Aging-resilient operation:** The control framework adapts naturally across lifetime without relying on fixed aging assumptions or recalibration-heavy maintenance.
- **Reduced guard-band dependence:** Direct in-situ observability, fast runtime response, and transient-aware mechanisms reduce pessimistic margin stacking across process variation, thermal behavior, supply noise, and aging.

4.3 Runtime Operational Model

The runtime operational model of VegaPower is organized around direct observation of margin-relevant behavior within the functional execution fabric, runtime interpretation of silicon state, and coordinated actuation of operating conditions to maintain correctness while improving efficiency. Unlike static OPP frameworks, which select among pre-approved operating points, or proxy-based methods, which infer behavior through indirect correlation, VegaPower ties the control loop directly to the timing behavior that matters during live execution.

At the foundation of this model is the use of **in-situ** observation on selected critical paths within the ASIC design through **Functional Path Monitors** (FPMs). Rather than relying on distributed proxies or generic monitor structures, the architecture is built around functional-path awareness within the real silicon fabric. This is important because the true timing limit of the chip is ultimately determined by the behavior of real paths under real workload conditions, not by abstracted estimates of silicon speed.

A key architectural advantage is that the solution is not restricted to a fixed and narrow set of paths selected only at initial deployment. The technology enables selection of relevant critical paths across the entire chip lifecycle, allowing the monitored path set to evolve as needed to address process shift and variation, workload-dependent path sensitization, thermal behavior and history, and long-term aging. This provides broader and more flexible coverage than approaches that depend on a small static set of presumed representative paths. As process characteristics shift, workloads change, and aging reorders path criticality over time, the observability framework remains aligned with the actual timing-limiting behavior of the silicon.

Within this model, runtime operation can be understood as a continuous sequence. First, the architecture observes margin-relevant behavior on selected in-situ paths during functional execution. Second, it interprets that behavior to determine the instantaneous runtime condition of the silicon, including how workload, local operating conditions, and lifetime effects influence available timing margin. Third, it coordinates actuation of voltage, frequency, and related timing-control mechanisms so that operating conditions remain aligned with real silicon capability rather than broad pre-characterized assumptions. This coordination is carried out by the **Adaptive Silicon Controller** (ASC), a synthesizable logic block.

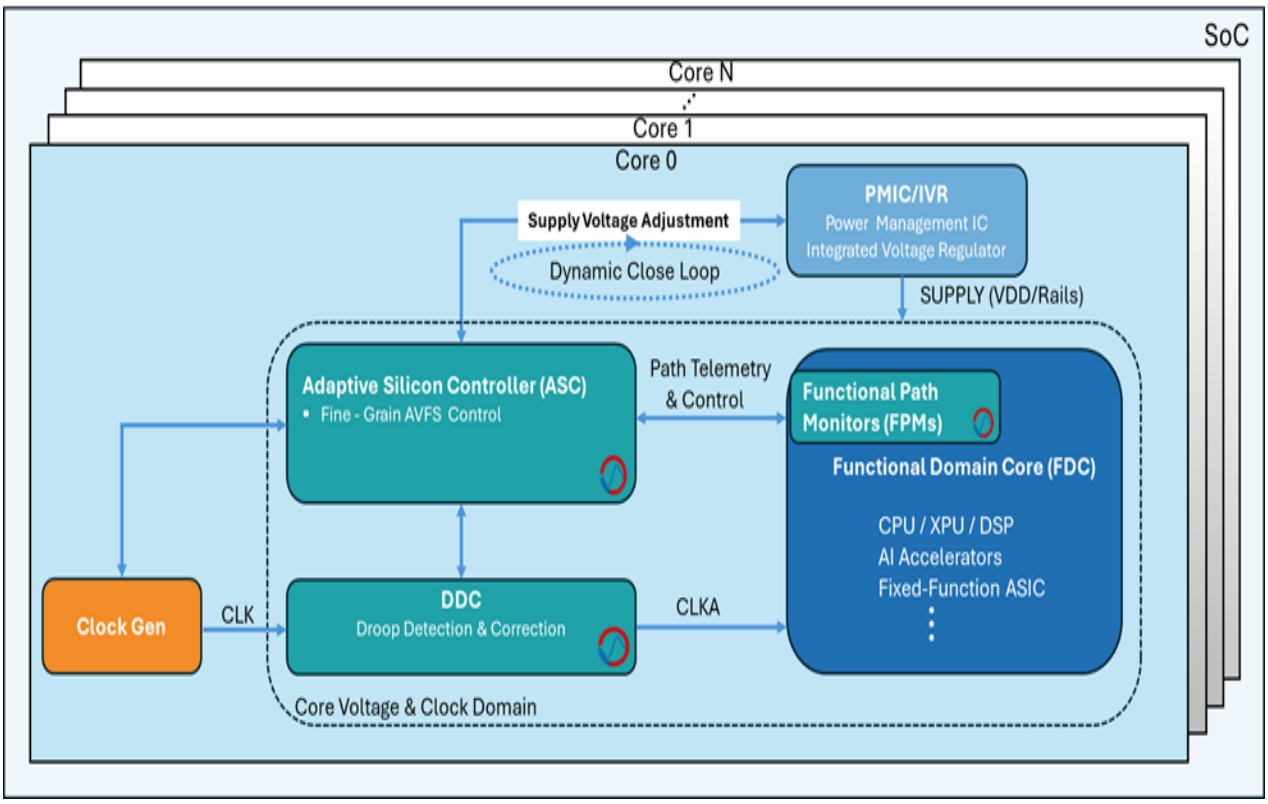


Figure 1 – VegaPower™ Runtime Power Architecture.

The architecture is organized around in-situ observation of selected critical paths within the functional ASIC fabric, runtime interpretation of silicon state, and coordinated actuation of operating conditions. Path selection can evolve across the chip lifecycle to maintain broad and flexible coverage under process variation, workload evolution, thermal, and aging.

In addition to continuous runtime regulation based on in-situ monitoring, the architecture can incorporate fast **droop detection and correction** (DDC) as a custom circuit within the power-management control loop. This is important because transient supply droops are a major source of guardband in conventional systems. Rather than absorbing droop uncertainty primarily through static voltage margin, VegaPower can address fast droop events directly during operation, allowing droop-related margin to be reduced through runtime correction rather than worst-case overprovisioning. In this sense, DDC operates as a complementary fast-response mechanism within the broader execution-coupled runtime architecture. VegaPower encompasses two coordinated loops: a slower loop that manages lower-varying conditions such as temperature, thermal behavior, and aging, and a fast loop running in parallel with sufficient bandwidth to detect and correct fast transient supply droops. This two-loop mechanism is depicted in Figure 2.

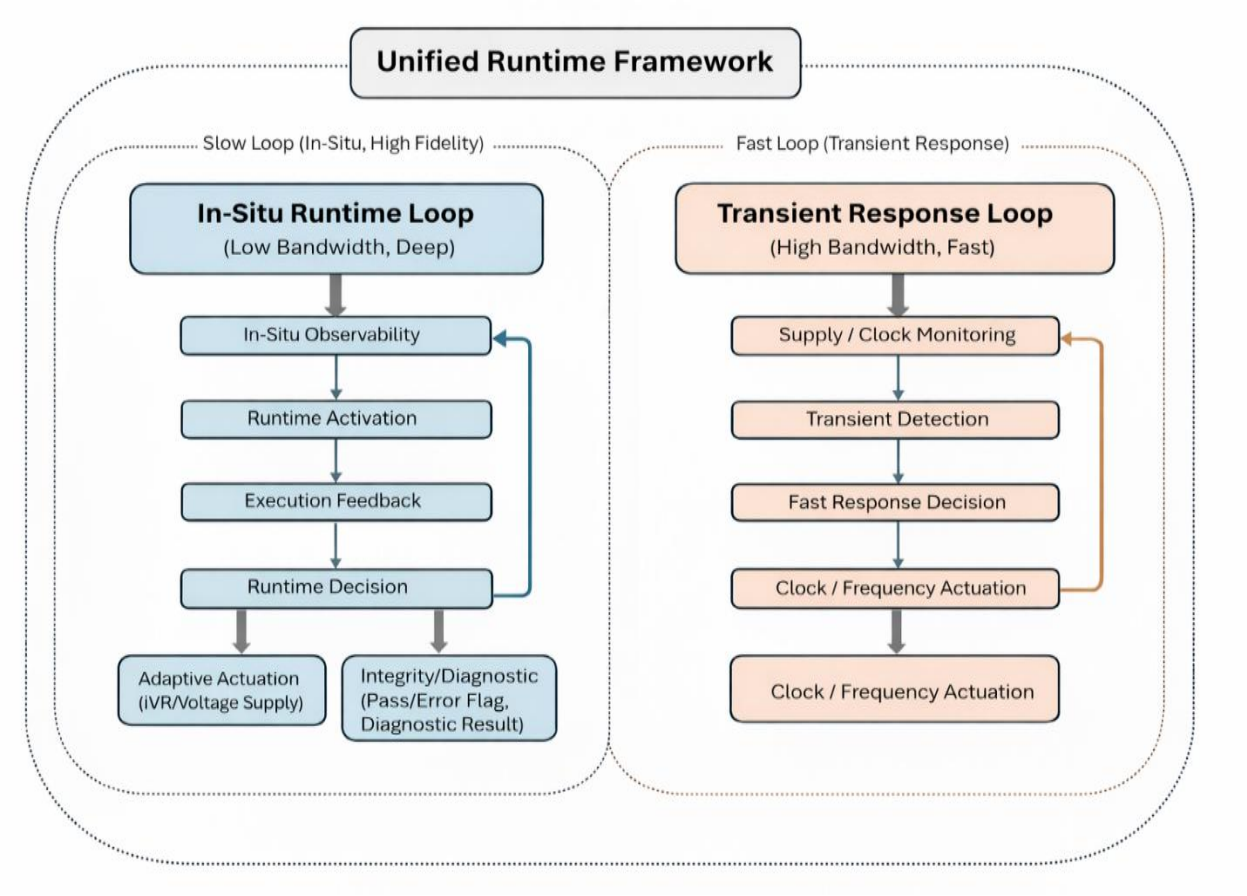


Figure 2 – VegaPower™ Fast and Slow Runtime Control Loops.

The VegaPower control model combines a slow execution-coupled loop that tracks workload, thermal behavior, and aging through in-situ critical-path observation with a fast transient loop that uses droop detection and correction (DDC) to mitigate rapid supply disturbances. Together, these loops reduce reliance on static guardbands while preserving safe operation.

5 Architectural Comparison with Existing Approaches

Key Observations

- Static OPP remains the most widely adopted power-management approach, offering implementation simplicity but no direct visibility into true runtime timing margin.
- RO- and replica-based AVFS improve runtime awareness, but remain indirect and correlation-dependent, with limited ability to track true functional-path behavior.
- VegaPower introduces direct execution-coupled observability, closed-loop runtime control, fine-grain adaptation, and reduced dependence on pre-characterization.
- The key comparison axis is not simply whether an approach supports AVFS, but how directly and robustly it manages real functional timing margin and how much residual guardband remains in the system.

Power management in modern SoCs is not only a control problem; it is also an architectural and economic one. The underlying power-management approach influences energy efficiency, silicon area, PDN and PMIC design, thermal stress, yield, product lifespan, and overall system cost. Accordingly, the comparison in this section is not limited to whether an approach can adjust voltage and frequency at runtime, but to how directly it captures true functional timing margin, how much uncertainty it preserves, and how that uncertainty propagates into the rest of the platform.

5.1 Comparison Framework

The most useful comparison is not simply whether an architecture supports DVFS or AVFS, but what it actually observes, how directly that observable represents true functional timing margin, how quickly the control loop can respond, and how much uncertainty remains after control is applied. Those factors determine not only runtime efficiency, but also how much conservatism must still be carried across voltage margin, PDN design, thermal management, yield, and product lifespan.

Under that framework, conventional OPP provides operational simplicity but essentially no direct runtime margin visibility. RO- and replica-based approaches improve runtime awareness, yet do so through indirect or correlation-dependent signals. VegaPower, by contrast, is defined by direct execution-coupled observation in the functional fabric and closed-loop control at execution-relevant time scales. The architectural distinction is therefore not incremental refinement of conventional AVFS, but a more direct and robust way of managing real timing margin during live operation.

5.2 Comparative Table

Dimension	Static Operating Points	RO / Replica-Based AVFS	VegaPower
Margin visibility	None at runtime	Indirect / proxy-based	Direct, execution-coupled observation
Relationship to real functional timing	Pre-characterized assumption	Correlation-dependent approximation	Directly tied to functional fabric behavior
Control model	Open-loop	Semi-closed loop	Closed-loop, execution-coupled
Response time scale	Design-time only / coarse runtime state change	Slow, averaged, correlation-limited	Execution-relevant time scales
Control granularity	Discrete, fixed operating points	Coarse AVFS steps	Fine-grain, continuous adaptation
Workload awareness	None	Weak / indirect	Inherent through functional execution
Thermal / hotspot awareness	Coarse and lagging	Limited / indirect	Directly reflected through in-situ path behavior
Aging Management	Worst-case at design	Model- and calibration-dependent	Naturally adaptive over lifetime
Early-failure guarantee	N/A	Non-deterministic	Not dependent on ordering guarantees
Characterization dependence	Heavy	Heavy	Reduced / autonomous runtime operation
Correlation drift risk	High	High over lifetime	Structurally minimized
Guard-band dependence	High	Reduced but residual	Fundamentally reduced
System efficiency	Systematically pessimistic	Moderately improved	Maximized within safe bounds

This comparison is central because it shows VegaPower improving multiple architectural dimensions simultaneously, not merely a single control-policy metric. The differentiation is not simply that it adapts more often, but that it is built on a more direct relationship between observability, timing margin, and runtime control.

6 System-Level Benefits of VegaPower

Key Observations

- Direct runtime margin control reduces the need for worst-case voltage inflation.
- Margin tracks real workload behavior and aging rather than static assumptions.
- PDN, PMIC, packaging, and thermal design can all benefit from reduced uncertainty.
- Yield distribution and lifetime consistency improve when runtime adaptation softens rigid static limits.

Power management in modern SoCs is not a narrow optimization problem. It is a multi-dimensional architectural challenge that affects power efficiency, silicon area, yield, thermal behavior, packaging, product lifespan, and overall system economics. The consequences of conservative margining extend well beyond excess voltage, shaping infrastructure decisions across PDN design, regulator capability, cooling requirements, and silicon utilization. As a result, a more accurate runtime power architecture creates value not only through lower power consumption, but through broader improvements in design efficiency, manufacturability, and lifecycle performance.

6.1 Power Efficiency

The most immediate benefit of VegaPower is improved power efficiency. By directly observing timing margin in situ within the functional fabric and regulating operation during execution, the architecture reduces dependence on worst-case voltage inflation and broad static guardbands. The result is lower dynamic power and leakage, achieved through tighter alignment between runtime silicon behavior and operating control.

6.2 Timing Margin Utilization

A second benefit of VegaPower is more effective utilization of actual timing margin. Rather than treating margin as a fixed reserve derived from broad characterization, the architecture allows available headroom to be evaluated in relation to real workload conditions, local silicon behavior, and aging over time. This enables safe use of margin that would otherwise remain trapped inside conservative assumptions, improving silicon efficiency across the product lifetime without relying on persistent worst-case pessimism.

6.3 PDN, PMIC, and Packaging Implications

VegaPower's impact extends beyond voltage selection into the surrounding power-delivery infrastructure. PDN overdesign for rare transients can be reduced because fast on-chip mitigation, including droop detection and correction, lowers dependence on broad PDN headroom. PMIC or IVR transient response requirements can also be relaxed because timing-critical control moves closer to the silicon execution paths themselves. Packaging and thermal

stress benefit as well, since tighter runtime control can reduce peak current demand and limit worst-case excursions that would otherwise force more conservative infrastructure choices.

6.4 Thermal and Lifetime Benefits

Continuous adaptation improves long-term operating consistency. Lower worst-case excursions reduce thermal stress, while runtime control that adapts continuously over time improves behavior across aging. This is important because static approaches often carry end-of-life pessimism from the beginning of product life, whereas VegaPower adapts with the silicon rather than against it. By reducing unnecessary overvoltage and reacting more directly to real conditions, the architecture can also help reduce cumulative thermal stress that would otherwise negatively affect product lifespan.

6.5 Yield and Productization Benefits

Runtime margin adaptation can soften rigid pass/fail limits and improve effective yield distribution and binning. When control is tied more closely to live silicon conditions, useful headroom can be preserved in devices that would otherwise be forced into overly conservative bins or operating states. Productization benefits therefore extend beyond watts into manufacturability, commercialization efficiency, and more effective utilization of silicon capability across the product portfolio.

6.6 System-Level Impact Table

Impact Dimension	What Changes with VegaPower	Why
Power efficiency	Reduced worst-case voltage inflation	Margin is observed and controlled during execution
Timing margin utilization	Margins track real workload and aging	Execution-coupled observability replaces static assumptions
PDN design	Reduced overdesign for rare transients	Fast on-chip mitigation lowers PDN headroom dependence
PMIC / IVR	Relaxed transient response requirements	Timing-critical control moves into silicon paths
Transient / clock integrity	Reduced global throttling and recovery	Localized, execution-aware correction
Area overhead	Reduced worst-case-driven logic area, decap insertion, and sensor/monitor overhead	Less uncertainty-driven over-provisioning
Yield & binning	Improved effective yield distribution	Runtime adaptation softens hard pass/fail limits
Packaging & thermals	Lower peak current and thermal stress, simplifies IVR and power-delivery design, fewer regulator phases, and reduced package BOM and integration overhead	Tighter control reduces worst-case excursions, lower thermal burden, and simplify PMIC subsystems
Product lifetime	More consistent behavior across aging and reduced thermal stress, increase hardware lifespan	Control adapts continuously over time

7 Integration and Deployment Path

Key Observations

- Industry resistance to runtime power control is driven as much by deployment risk as by technical skepticism.
- OPP-based frameworks persist because they are familiar, validated, and easy to qualify, not because they fully solve modern power-management needs.
- VegaPower can be introduced in stages alongside existing OPP infrastructure rather than as an abrupt replacement.
- A staged deployment path allows confidence to be built through observation, correlation, controlled refinement, and progressive runtime authority.
- This approach reduces adoption risk while creating a practical path toward more direct runtime power control.

A key practical challenge in modern power management is not only identifying a better runtime architecture, but introducing it in a way that aligns with established product-validation and deployment practices. Power control remains one of the most risk-sensitive functions in SoC design, and industry reliance on OPP-based frameworks reflects not only technical history, but the need for predictable qualification, debug visibility, and operational confidence. For this reason, the transition to a more direct runtime power architecture must be achievable through staged integration rather than all-at-once replacement.

7.1 Why Deployment Resistance Exists

Industry resistance to new runtime power methods is not irrational. Voltage and frequency control sit close to the boundary between efficiency and failure, and mistakes directly affect performance, correctness, thermal behavior, and product reliability. For that reason, organizations have historically preferred pre-characterized operating frameworks such as OPP, even when those frameworks are known to be pessimistic. Their appeal lies in validation familiarity, qualification history, and the relative ease of explaining system behavior through bounded operating states.

This resistance is reinforced by the history of runtime techniques themselves. Many prior approaches have relied on indirect proxies, fragile long-term correlation, limited coverage, or mechanisms that are difficult to validate comprehensively under real workloads. As a result, the industry has often viewed runtime methods as interesting in principle but difficult to trust in product deployment. In practice, the barrier is therefore not only whether a runtime architecture can improve efficiency, but whether it can be introduced with acceptable risk and measurable confidence.

7.2 Parallel Operation with Existing OPP Frameworks

For this reason, VegaPower is best introduced initially as a runtime architecture that can operate alongside the existing OPP and governor framework rather than replacing it immediately. In the first stage of deployment, the conventional OPP control path remains the active operating mechanism, while VegaPower runs in parallel as an observational and evaluative layer. In this role, VegaPower monitors in-situ margin-relevant behavior in the functional fabric and assesses how actual silicon conditions compare with the assumptions embedded in the pre-characterized operating envelope.

This mode is important because it allows the architecture to be introduced without changing the system's primary power-control authority. Existing product behavior, safety margins, validation flows, and qualification assumptions remain intact, while VegaPower begins building a runtime understanding of real silicon behavior under field-relevant workloads. In effect, the architecture can initially serve as a truth-extraction layer, revealing where conventional OPP assumptions remain overly conservative, where workload-specific stress diverges from pre-characterized expectations, and where additional runtime intelligence can create value.

7.3 Guided Runtime Refinement Within the Existing Envelope

Once VegaPower has been deployed in parallel and sufficient confidence has been built, the next stage is to use it as a guided runtime refinement layer within the existing OPP framework. At this stage, the architecture does not yet fully replace the conventional operating model, but it begins to influence how the pre-approved envelope is used. Existing governors and OPP states remain present, yet their selection and interpretation are increasingly informed by direct in-situ observability rather than only by indirect telemetry or static policy rules.

This stage creates a practical middle ground between conservative deployment and full runtime-native control. VegaPower can identify where the current operating state carries excess margin, where workload-specific conditions justify tighter control, and where droop- or thermally driven events are being handled too pessimistically by the existing framework. In this way, the architecture begins to unlock efficiency and infrastructure benefits without requiring the organization to abandon the validation comfort of a pre-characterized envelope all at once.

7.4 Progressive Expansion of Runtime Authority

As deployment maturity increases, VegaPower can move from guided refinement toward more direct runtime authority over operating conditions. In this stage, the architecture plays a larger role in coordinating voltage, frequency, and transient-response decisions based on execution-coupled observation of real functional behavior. Static OPP states no longer remain the primary source of intelligence; instead, they increasingly become a bounded baseline, fallback layer, or qualification anchor around a more capable runtime control system.

This transition is especially important because it enables the full architectural value of VegaPower to emerge. Once runtime observability is allowed to influence power control more

directly, conservative guardbands can be reduced more systematically, margin can be managed as a live variable rather than a fixed reserve, and mechanisms such as fast droop detection and correction can begin replacing portions of the static margin historically carried to cover transient uncertainty. At that point, the architecture moves from coexistence with legacy control toward runtime-centric management of real silicon capability.

7.5 Why a Staged Path Matters

A staged deployment path is important because it aligns architectural ambition with the realities of product qualification and organizational risk. VegaPower does not need to be introduced as an abrupt replacement for existing OPP-based power management. Instead, it can be adopted through a progression that mirrors how the industry builds trust in new control mechanisms: first observe, then refine within a bounded envelope, and only then assume broader runtime authority.

This approach reduces adoption risk in several ways. It preserves debug visibility, supports incremental qualification, allows runtime behavior to be evaluated under real applications, and avoids the organizational anxiety associated with all-or-nothing deployment. It also creates a measurable path for demonstrating value, since each stage can expose evidence of excess static margin, limited proxy fidelity, droop-related pessimism, and unrealized silicon headroom. In that sense, the deployment path is not separate from the architecture; it is part of how the architecture becomes credible.

A further advantage is that this staged model aligns naturally with the lifecycle-aware nature of VegaPower itself. The architecture is built to observe selected critical paths across the product lifetime, allowing monitored coverage and control relevance to evolve as process realization, workload behavior, thermal stress, and aging reshape the timing profile of the silicon. This makes the deployment path more robust over time than methods that require confidence to be established only once and then assumed to remain valid indefinitely.

For that reason, staged deployment is not merely an implementation convenience or commercialization detail. It is part of the architectural strategy. It provides a credible bridge between today’s pre-characterized power-management practice and a future in which runtime control is driven by direct in-situ observation of the functional fabric itself.

8 Why This Matters Now

Key Observations

- Advanced nodes are reducing physical operating margin while increasing runtime uncertainty.
- Modern workloads amplify localized timing, thermal, and power-delivery stress.
- Conservative guard-banding now propagates into area, infrastructure cost, yield, and product lifespan.
- Power management is becoming a foundational architectural problem rather than a secondary optimization layer.

The need for a new runtime power architecture is becoming stronger, not weaker. As semiconductor technologies continue to advance through FinFET, GAAFET, and future device generations, the traditional margin available to absorb uncertainty keeps shrinking. At the same time, non-uniform variation, localized thermal behavior, interconnect sensitivity, and lifetime degradation make real operating conditions more difficult to predict and manage through design-time assumptions alone.

Modern application demand intensifies this challenge. AI workloads, heterogeneous compute fabrics, bursty accelerators, and rapidly shifting execution phases create highly dynamic activity patterns that amplify localized current demand, hotspot formation, and timing stress. These workloads do not merely run on advanced-node silicon; they increasingly expose and magnify its underlying sensitivities. Under such conditions, power management can no longer rely primarily on static operating assumptions or indirect runtime approximation without carrying substantial residual uncertainty.

That uncertainty now has a broader cost than in earlier generations. What begins as conservative timing margin becomes excess voltage, elevated power, greater thermal stress, stronger PDN and PMIC requirements, more difficult packaging tradeoffs, and reduced effective use of silicon capability. It can also negatively affect yield opportunity and product lifespan by forcing the system to operate with unnecessary pessimism over long periods of time. In this sense, guard-banding is no longer just a circuit-level safety measure; it has become a system-level economic burden.

For this reason, power management must now be viewed as a first-order architectural discipline. The relevant question is no longer whether runtime control should exist, but what kind of runtime control can remain accurate, scalable, and economically meaningful under modern silicon conditions. VegaPower addresses that need by moving power management away from pre-characterized assumptions, lagging indicators, and indirect proxies toward direct in-situ observation of real behavior in the functional fabric. That shift is timely because the underlying industry pressures are no longer incremental. They are structural, and they are accelerating.

9 Conclusion

Modern SoC power management is approaching the limit of what can be achieved through static operating assumptions, indirect proxies, and increasingly complex policy refinement. At advanced technology nodes, device-physics challenges such as non-uniform process variation, thermal confinement, slow heat diffusion, severe aging effects, and growing interconnect sensitivity make timing behavior more localized, dynamic, and difficult to infer reliably. Under these conditions, OPP-based control and proxy-driven runtime techniques, even when aided by temperature sensors and other telemetry, remain fundamentally limited because the information they rely on is indirect, spatially incomplete, and often lagging relative to the real timing stress developing in the functional fabric.

VegaPower addresses this challenge through a different architectural approach: direct in-situ observation of the functional fabric, closed-loop runtime control, lifecycle-flexible path coverage, and fast transient response. By shifting power management from estimation-driven guardbanding to execution-coupled runtime regulation, the architecture enables a more scalable path toward improved efficiency, reduced overdesign, better silicon utilization, and stronger lifetime behavior.

In this sense, VegaPower is not merely an enhancement to existing AVFS practice. It represents a broader transition toward runtime power intelligence as a foundational capability for modern SoC design.