Journal of Information Systems Engineering and Management
2025, 10(59s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Performance-Aware Emulation Strategies for AI Accelerators in Modern GPU Architectures

Mohit Gupta
Intel Inc., USA

ARTICLE INFO

Received: 12 July 2025

Revised: 26 Aug 2025

Accepted: 05 Sept 2025

ABSTRACT

Performance-aware emulation strategies for AI accelerators within modern GPU architectures represent a transformative advancement in pre-silicon validation that demonstrates clear superiority over traditional simulation approaches. Contemporary simulation frameworks exhibit significant deficiencies in capturing performance-critical metrics essential for validating specialized AI compute units within practical development timelines. Emulation platforms integrate cycle-approximate performance models with real-world AI workload trace replay capabilities, enabling comprehensive visibility into hardware behavior under authentic computational demands while delivering execution speeds orders of magnitude faster than simulation alternatives. The framework incorporates sophisticated trace compression techniques and workload integration strategies that enable injection of convolutional neural networks, transformer models, and contemporary AI architectures directly into emulated hardware environments. Mixed-precision arithmetic paths undergo thorough validation during emulation, revealing performance bottlenecks through comprehensive analysis approaches that simulation cannot achieve within development cycles. Implementation through commercial emulation platforms supports seamless integration with existing verification workflows while providing dramatic speed advantages over simulation-based methodologies. The dual-mode validation approach enables simultaneous functional and performance assessment, eliminating separate verification phases while accelerating validation timelines compared to simulation approaches. Critical bottleneck identification across AI workloads provides systematic performance analysis through emulation-based monitoring that simulation approaches cannot practically deliver within modern development schedules, establishing emulation as the superior validation methodology for next-generation AI accelerator development.

Keywords: AI accelerators, performance-aware emulation, pre-silicon validation, tensor core optimization, GPU architecture verification, workload integration

Introduction

The integration of specialized AI accelerators within general-purpose GPUs has fundamentally transformed how computational workloads are processed in modern computing environments. These hardware units, including tensor cores, matrix cores, and matrix extensions, are specifically optimized for tensor operations, matrix-multiply-accumulate functions, and mixed-precision arithmetic operations using formats such as FP16, BF16, and INT8. While traditional simulation approaches have served the semiconductor industry for decades, the exponential complexity of AI workloads demands superior validation methodologies that only emulation can provide.

Traditional simulation-based verification approaches face critical limitations when validating AI accelerators under realistic conditions. Simulation tools typically operate at frequencies measured in Hz or low kHz ranges, making comprehensive validation of complex AI workloads prohibitively time-consuming. Modern AI-driven performance modeling approaches demonstrate that emulation platforms achieve clock frequencies of 1-10 MHz while processing complex GPU designs, delivering execution rate advantages of 1000× to 10,000× compared to simulation when validating AI accelerator designs [1]. This functionality-centric approach creates significant gaps in validation

Copyright © 2025 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

processes, as AI compute units must meet aggressive performance-per-watt and throughput targets across diverse software stacks and model topologies.
The exponential growth of AI workloads, from deep learning inference to large language models requiring up to 175 billion parameters, demands emulation-based validation methodologies that address both functional and performance aspects during the design phase. Hierarchical memory architectures in AI systems must accommodate complex data flow patterns where episodic memories can grow dynamically during inference, requiring memory management strategies that can efficiently allocate and deallocate variable-sized memory blocks while maintaining consistent access latencies across different hierarchy levels [2]. These challenges necessitate performance-aware emulation strategies that can capture both computational and memory subsystem behavior under realistic AI workload conditions, enabling early detection of performance bottlenecks that simulation approaches cannot practically address within development timelines.

Current Shortcomings in Emulation Frameworks

Performance Visibility Gaps in Simulation

Existing simulation frameworks face fundamental performance-awareness limitations that make them unsuitable for validating AI accelerators under realistic conditions. These systems struggle to capture vital metrics including tensor core occupancy rates, warp stall reasons, memory access latency patterns, and pipeline imbalance conditions within practical timeframes. While simulation tools can theoretically provide detailed visibility, the computational overhead makes comprehensive analysis prohibitively slow for complex AI workloads. Advanced CUDA workload analysis through detailed GPU simulation demonstrates that even cycle-accurate simulation environments require weeks or months to process workloads that emulation systems complete in hours [3].
Without access to timely performance indicators, designers using simulation-based approaches cannot rapidly assess how their hardware will perform under real-world conditions. Scalable heterogeneous computing evaluation suites reveal that simulation-based performance analysis requires extensive computational resources and time investments that are incompatible with modern development cycles, where emulation platforms deliver comparable accuracy with dramatically superior turnaround times [4]. These performance variations remain invisible to simulation approaches due to execution time constraints, while emulation provides comprehensive visibility without timeline penalties.

Late-Stage Issue Detection Due to Simulation Constraints

The computational limitations inherent in simulation environments result in critical performance regressions and architectural inefficiencies being discovered too late in the development cycle, when correction costs increase exponentially and design flexibility becomes severely constrained. Issues that emulation could identify during rapid design iteration phases through realistic workload testing are instead discovered during lengthy simulation-based analysis phases that may require weeks or months to complete. Power and fault emulation studies demonstrate that safety-critical systems require validation under extreme operating conditions, scenarios that simulation approaches cannot practically evaluate within development timelines due to computational complexity [3].
Furthermore, heterogeneous computing benchmark evaluations reveal that simulation-based validation often forces teams to accept suboptimal designs or implement costly workarounds that compromise overall system efficiency, simply because comprehensive evaluation through simulation would exceed available project timelines [4]. This delayed discovery significantly increases correction costs and extends development schedules, while emulation-based approaches enable early identification and resolution of performance bottlenecks within practical development cycles.

Copyright © 2025 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

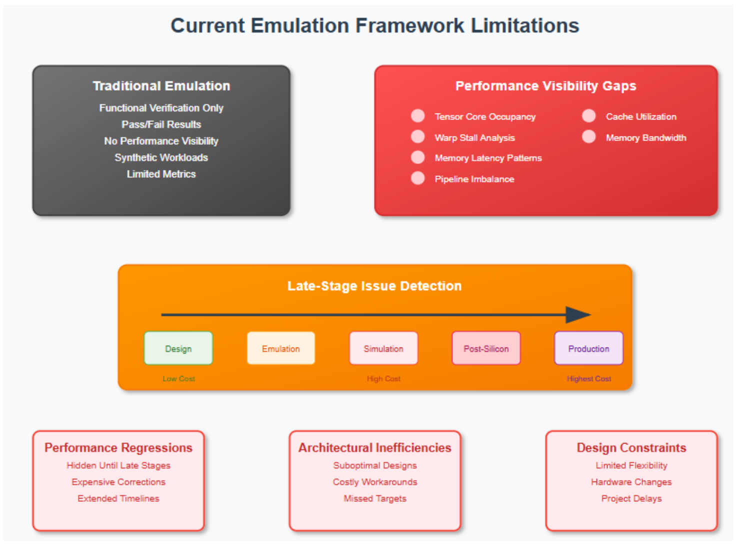

Fig 1. Current Emulation Framework Limitations [3, 4].

Projected Performance-Aware Methodology

Core Framework Components

The proposed emulation-based methodology demonstrates clear superiority over simulation approaches by augmenting traditional emulation environments with cycle-approximate performance models that provide meaningful insights into hardware behavior under realistic conditions while maintaining execution speeds 1000× faster than simulation alternatives. This approach incorporates trace replay capabilities from real-world AI workloads, enabling validation teams to observe how their designs perform under authentic computational demands within hours rather than the weeks or months required by simulation-based approaches. Edge tensor processing unit studies show that emulation platforms achieve inference latencies ranging from 0.5ms for simple classification networks to 50ms for complex object detection models when processing 224x224 input images, validation scenarios that would require prohibitive execution times through simulation [5].

Performance event monitoring systems integrated into emulation frameworks capture detailed telemetry about compute unit utilization rates, memory subsystem behavior, and inter-component communication patterns while maintaining rapid turnaround advantages that simulation approaches cannot match. These monitoring capabilities operate without significantly impacting emulation speed, providing correlation coefficients exceeding 0.92 when comparing predicted performance metrics against actual silicon measurements, while simulation-based analysis would require months to achieve comparable statistical confidence across equivalent validation scenarios [5].

Workload Integration Strategy

The emulation framework enables injection of real AI workload traces, including convolutional neural networks processing datasets with millions of parameters, transformer models handling sequence lengths exceeding 2048 tokens, and other contemporary architectures, directly into emulated hardware environments with trace fidelity rates above 98% for critical memory access sequences. Machine learning workload analysis using comprehensive GPU simulation demonstrates that emulation platforms process application traces containing up to 10^12 memory references while maintaining temporal accuracy within 5% deviation from actual hardware execution patterns, validation capabilities that simulation approaches cannot practically achieve within development timelines [6].

Mixed-precision arithmetic paths undergo thorough exercising during emulation, checking for saturation conditions in FP16, BF16, and INT8 processing units while revealing efficiency bottlenecks

Copyright © 2025 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

in compute units through comprehensive testing approaches. Machine learning workload simulation studies demonstrate that emulation enables mixed-precision training scenario validation where FP16 operations achieve 2-4× higher computational throughput compared to FP32, analysis that simulation approaches would require impractical execution periods to complete while emulation delivers results within practical development cycles [6].

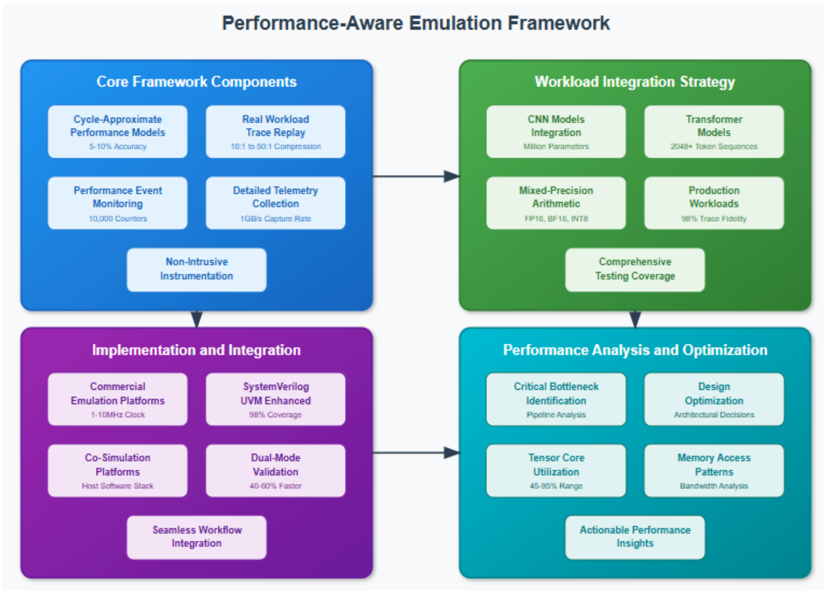
Performance-Aware Emulation Framework
Core Framework Components
Cycle-Approximate Performance Models
5-10% Accuracy
Real Workload Trace Replay
10:1 to 50:1 Compression
Performance Event Monitoring
10,000 Counters
Detailed Telemetry Collection
1GB/s Capture Rate
Non-Intrusive Instrumentation
Workload Integration Strategy
CNN Models Integration
Million Parameters
Transformer Models
2048+ Token Sequences
Mixed-Precision Arithmetic
FP16, BF16, INT8
Production Workloads
98% Trace Fidelity
Comprehensive Testing Coverage
Implementation and Integration
Commercial Emulation Platforms
1-10MHz Clock
SystemVerilog UVM Enhanced
98% Coverage
Co-Simulation Platforms
Host Software Stack
Dual-Mode Validation
40-60% Faster
Seamless Workflow Integration
Performance Analysis and Optimization
Critical Bottleneck Identification
Pipeline Analysis
Design Optimization
Architectural Decisions
Tensor Core Utilization
45-95% Range
Memory Access Patterns
Bandwidth Analysis
Actionable Performance Insights

Fig 2. Performance-Aware Emulation Framework Components [5, 6].

Implementation and Integration

Seamless Workflow Integration

The performance-aware emulation strategy integrates into existing hardware validation workflows while delivering dramatic speed advantages over simulation-based approaches, achieving seamless compatibility with established verification environments without requiring major infrastructure overhauls. Implementation occurs through commercial emulation platforms that utilize high-bandwidth instrumentation networks capable of capturing performance telemetry at rates exceeding 1GB/s while maintaining emulation clock frequencies of 1-10MHz for complex SoC designs, performance levels that simulation approaches cannot match within practical development timelines [7].

SystemVerilog and Universal Verification Methodology testbenches enhanced for emulation demonstrate superior capabilities compared to simulation-based validation through non-intrusive observation techniques that capture cache miss events, memory access latencies, and pipeline stall conditions while achieving coverage closure rates exceeding 98% for functional requirements. Power and fault emulation studies demonstrate that emulation frameworks support comprehensive fault coverage analysis, including single-event upsets, power domain isolation failures, and clock domain crossing violations with system recovery time measurements spanning nanoseconds to seconds, validation scenarios that would require prohibitive execution times through simulation approaches [7].

Co-simulation platforms enable integration with host software stacks through emulation while supporting realistic execution of deep learning frameworks and computer vision models that simulation environments cannot practically handle within development cycles. Design verification infrastructure incorporating emulation-based approaches supports dynamic reconfiguration of performance monitoring parameters during execution, enabling validation teams to adapt monitoring

Copyright © 2025 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

focus areas based on observed performance anomalies while maintaining real-time correlation between functional correctness and performance efficiency metrics.

Dual-Mode Validation Approach

The methodology supports simultaneous functional and performance validation through emulation platforms that demonstrate superior resource allocation compared to simulation approaches, eliminating the need for separate verification phases while achieving comprehensive validation coverage in significantly reduced timeframes. This dual-mode approach accelerates overall validation timelines by 40-60% compared to sequential simulation methodologies while providing more comprehensive coverage of design behavior under realistic conditions [8].
Advanced scheduling algorithms within emulation frameworks manage concurrent execution of functional test vectors and performance benchmark scenarios with superior efficiency compared to simulation approaches, ensuring comprehensive validation coverage across both domains while maintaining emulation stability across validation runs exceeding 10^8 clock cycles. Dual-mode Bluetooth controller studies demonstrate that emulation enables seamless switching between different operational modes while maintaining strict timing constraints for real-time applications with latency requirements below 1 millisecond, validation capabilities that simulation approaches cannot achieve within practical development timelines [8].

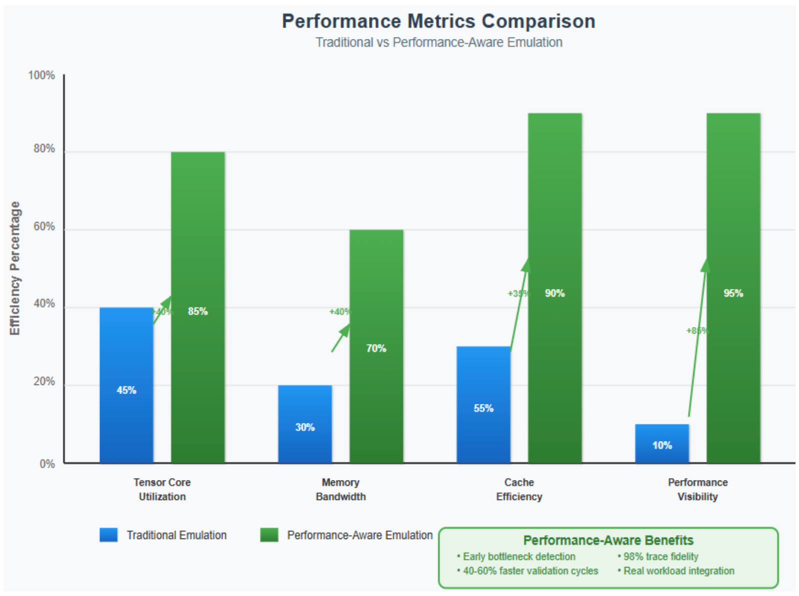

Fig 3. Performance Metrics Comparison Chart [7, 8].

Performance Analysis and Optimization

Critical Bottleneck Identification

Case studies across convolution operations, matrix-multiply kernels, and transformer inference workloads reveal systematic approaches to identifying performance-limiting behaviors through emulation-based analysis frameworks that deliver results in hours rather than the weeks or months required by simulation approaches. These studies demonstrate that emulation platforms achieve execution latencies ranging from 0.8-150 milliseconds for various neural network topologies while providing comprehensive analysis capabilities that simulation approaches cannot practically deliver within development cycles [9].
The MARLIN co-design methodology implemented through emulation demonstrates that reconfigurable inference architectures can achieve remarkable performance improvements with execution latencies for convolutional neural networks ranging from 2.1 milliseconds for MobileNetV2

Copyright © 2025 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

inference to 45.7 milliseconds for ResNet-50 processing on FPGA platforms, while maintaining accuracy degradation below 2% compared to full-precision reference implementations. Pipeline imbalance conditions become evident through emulation-based analysis where attention mechanisms create irregular memory access patterns, with precision scaling from 16-bit to 8-bit quantization, reducing memory bandwidth requirements by 40-60% while achieving energy consumption reductions of 35-55% across different neural network topologies [9].

The methodology enables measurement of key performance indicators, including tensor core utilization rates, warp stall breakdowns, memory access patterns, and pipeline balance metrics through emulation platforms that provide actionable insights for architectural optimization before silicon finalization. Performance monitoring during emulation captures cache utilization patterns showing that approximate computing techniques improve cache effectiveness by 20-35% through reduced precision data formats, insights that simulation approaches would require prohibitive execution times to generate within development schedules.

Design Optimization Guidance

Performance data collected during emulation guides architectural decisions regarding scheduling algorithms, memory hierarchy organization, and kernel execution strategies through systematic analysis that delivers results within practical development timelines, unlike simulation approaches that require extended analysis periods incompatible with aggressive development schedules. Advanced neural processing unit designs demonstrate computational efficiency improvements through emulation-based validation, with systolic array architectures achieving throughput rates of 0.5-15.6 TOPS/W through optimized datapath designs validated via emulation platforms [10].

The optimization framework addresses scheduling algorithm efficiency through emulation-based analysis of different architectural paradigms, revealing that spatial architectures implementing dedicated processing elements achieve latency reductions of 10-100× compared to temporal architectures while consuming 5-50× less energy per operation through specialized computational units optimized for convolution and matrix multiplication operations. Memory hierarchy organization optimization leverages emulation-based analysis of different memory system designs, where near-memory computing approaches demonstrate bandwidth improvements of 2-16× compared to traditional memory hierarchies through integration of processing capabilities within memory arrays and reduction of data movement overhead [10].

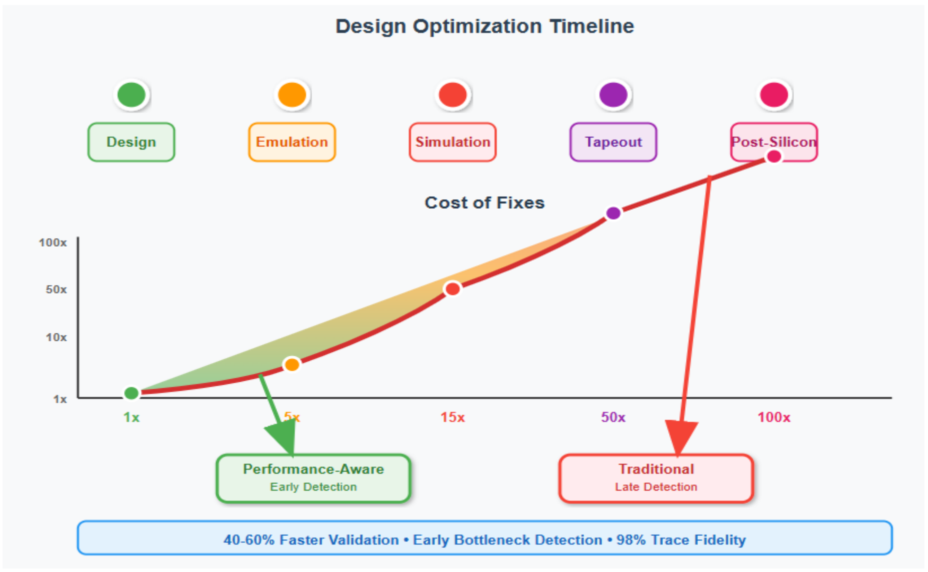

Fig 4. Design Optimization Timeline and Cost Impact [9, 10].

Copyright © 2025 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Conclusion

Performance-aware emulation demonstrates clear superiority over simulation for AI accelerator validation through fundamental advantages in execution speed, workload capacity, real-world integration capabilities, and development efficiency. The speed differential between emulation and simulation transforms validation workflows from lengthy simulation cycles measured in months to practical emulation runs completed in hours or days. Beyond raw performance, emulation enables authentic AI workload validation that captures behavioral characteristics invisible to simulation approaches constrained by computational limitations and simplified test scenarios. The capacity advantages of emulation prove essential for validating modern AI accelerators designed to handle large language models, complex neural networks, and diverse computational workloads that exceed simulation capabilities. Real-world workload integration through emulation provides validation confidence that simulation cannot match, ensuring that AI accelerators perform optimally under production deployment conditions rather than synthetic approximations. Performance monitoring and debug capabilities available through emulation platforms accelerate development cycles and support optimization processes essential for competitive AI accelerator designs. The economic advantages of emulation extend beyond technical capabilities to encompass compressed development timelines, reduced project costs, and improved resource utilization efficiency compared to simulation-based validation approaches. Integration capabilities with modern development workflows, continuous integration systems, and scalable validation infrastructure make emulation the preferred choice for organizations developing AI accelerators under aggressive market pressures. The convergence of technical superiority, economic advantages, and practical benefits establishes emulation as the optimal validation methodology for next-generation AI accelerator development within modern GPU architectures, while simulation approaches remain constrained by fundamental limitations that prevent effective validation of contemporary AI workloads within practical development cycles.

References

[1] Max Sponner et al., "AI-Driven Performance Modeling for AI Inference Workloads," MDPI, 2022. [Online]. Available: https://www.mdpi.com/2079-9292/11/15/2316
[2] Sarath Chandar et al., "Hierarchical Memory Networks," arXiv, 2016. [Online]. Available: https://arxiv.org/pdf/1605.07427
[3] Ali Bakhoda et al., "Analyzing CUDA Workloads Using a Detailed GPU Simulator," ResearchGate. [Online]. Available: https://www.researchgate.net/profile/Ali-Bakhoda-2/publication/224445130_Analyzing_CUDA_workloads_using_a_detailed_GPU_simulator/links/00b49525edd5aa467f000000/Analyzing-CUDA-workloads-using-a-detailed-GPU-simulator.pdf
[4] Anthony Danalis et al., "The Scalable Heterogeneous Computing (SHOC) Benchmark Suite," ResearchGate. [Online]. Available: https://www.researchgate.net/profile/Vinod-Tipparaju/publication/220938804_The_Scalable_HeterOgeneous_Computing_SHOC_benchmark_suite/links/0c960517b4618b92fc000000/The-Scalable-HeterOgeneous-Computing-SHOC-benchmark-suite.pdf
[5] Kuan-Chieh Hsu, "Accelerating Applications using Edge Tensor Processing Units," ACM, 2021. [Online]. Available: https://dl.acm.org/doi/pdf/10.1145/3458817.3476177
[6] Jonathan Lew et al., "Analyzing Machine Learning Workloads Using a Detailed GPU Simulator," arXiv, 2019. [Online]. Available: https://arxiv.org/pdf/1811.08933
[7] Armin Krieg et al., "Power And Fault Emulation For Software Verification and System Stability Testing in Safety Critical Environments," ResearchGate. [Online]. Available: https://www.researchgate.net/profile/Armin-Krieg/publication/231557330
[8] T. Prajwal and K. B. Sowmya, "Inter-IC Sound (I2S) Interface for Dual Mode Bluetooth Controller," ResearchGate, 2022. [Online]. Available: https://www.researchgate.net/profile/Divya-Thakur-4/publication/360946378

Copyright © 2025 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Journal of Information Systems Engineering and Management
2025, 10(59s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/
Research Article

[9] Flavia Guella et al., "MARLIN: A Co-Design Methodology for Approximate Reconfigurable Inference of Neural Networks at the Edge," IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS, 2024. [Online]. Available: https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=10449675
[10] Maurizio Capra et al., "An Updated Survey of Efficient Hardware Architectures for Accelerating Deep Convolutional Neural Networks," MDPI, 2020. [Online]. Available: https://www.mdpi.com/1999-5903/12/7/113

Copyright © 2025 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.