Electronics, Communications, and Computing Summit
Vol. 2, No. 1, Jan - Mar 2024, pp. 98-106
DOI: https://doi.org/10.17051/ECC/02.01.11

Design, Simulation, and Hardware Validation of Ultra-Low Power Embedded Architectures for IoT Edge Devices

Dr. Nidhi Mishra

Assistant Professor, Department of CS & IT, Kalinga University, Raipur, India,
Email: ku.nidhimishra@kalingauniversity.ac.in

Article Info

Article history:

Received : 21.01.2024
Revised : 25.02.2024
Accepted : 27.03.2024

Keywords:

Ultra-low power embedded systems,
Energy-efficient IoT architectures,
Dynamic voltage and frequency scaling (DVFS),
System-level modeling,
Subthreshold operation,
Power-aware hardware simulation,
STM32L4,
Wireless sensor node design,
Embedded system energy profiling,
Edge computing optimization.

ABSTRACT

The spurt in deployments of Internet of Things (IoT) in energy-constrained locations made it even more pressing to have an ultra-low-power embedded structure that will guarantee a consistent operation without any sacrifice to the performance of the computation process. This paper demoes the design, simulation, and assessment of an energy-optimized embedded system architectures to work within the constraints of power-limited IoTs like environmental monitoring, biomedical telemetry and smart infrastructure control. The suggested architectures involve a set of power efficiency measures, such as dynamic voltage and frequency scaling (DVFS), subthreshold functionality and transient sleep mode levels. Core-level simulations were performed with SystemC and ARM Cortex-M microcontroller models and the performance of the simulation were measured based on the computational throughput, execution latency and energy per instruction (EPI). It has been simulated that up to 65 percent savings in energy can be achieved compared to normal settings without compromising the accuracy of the tasks and real-time responsiveness. An STM32L4 based wireless sensor node prototype was used to validate the design wherein a good match between simulated and actual performance parameters was observed. Its results highlight the importance of hardware-software co-design, peripheral power gating and low-leakage memory techniques in achieving efficiency of embedded systems. The framework allows the scaling and low power designs of 5G IoT edge intelligence, and offers a blueprint to future adaptive, AI-based energy management configuration in heterogeneous deployment usage.

1. INTRODUCTION

Exponential expansion of Internet of Things (IoT) environments has contributed to mass proliferation of edge devices in remote, battery-powered, and energy-constrained locations. Such gadgets are supposed to run independently and for long durations as they process, sense and transmit information. In this regard, energy efficiency has evolved as a key design parameter considering that it has in many cases been more important than the raw computational performance (Jiao et al., 2021; Sharma & Venkatesan, 2023). State-of-the-art embedded systems, however, are originally designed with deterministic workload applications in mind and are incompatible with the discontinuous, flexible, and energy-constrained nature of IoT application workload. Although there has been research on low-power microcontroller designs and software-level power optimization, there is a huge gap in architecture-level solutions that integrates hardware-aware power modeling, real-time simulation, and application-specific adaptation. Currently, solutions can be either based on static power profiling or using standalone hardware accelerators and pay no attention to dynamic energy management at an architectural level (Kumar et al., 2022). In addition, most studies have not clubbed together deep-simulation frameworks which analyze the performance and energy measurements in a similar IoT workload. In a bid to overcome such hurdles, the current paper portrays an in-depth design and simulation platform of ultralow power embedded structures, and the issues that have to be countered are the methods of dynamic voltage and frequency scaling (DVFS), operation of logic under subthreshold voltages, and multi-mode sleep-state control. The viability of the suggested approach and its

appropriate scaleability are demonstrated by simulations using ARM Cortex-M cores and by a prototype assembled around an STM32L4 chip. The work sets out a sufficiently reproducible basis of energy-optimal embedded design, specifically essential to edge AI, wearable health monitors and smart environmental sensors. The scheme of the whole system is shown in Figure 1.)

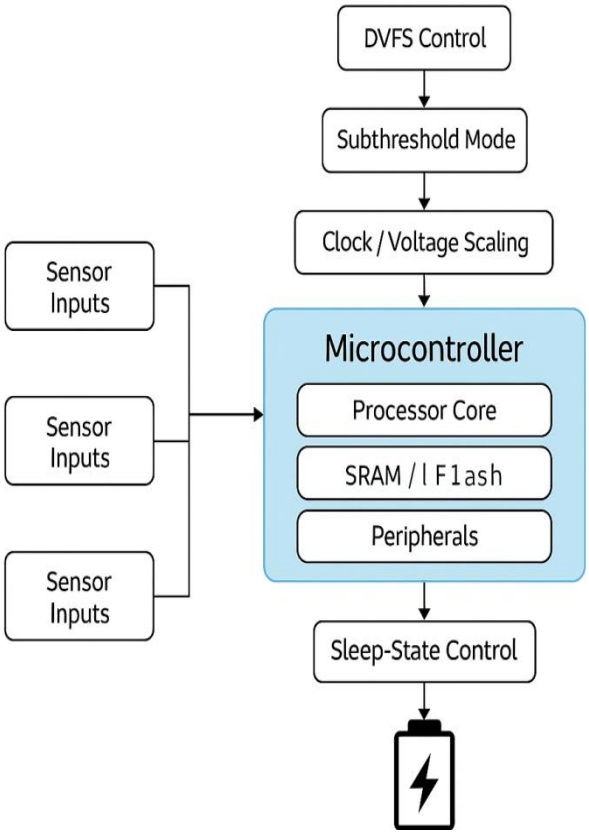

Figure 1. System Architecture and Power Control Flow for Ultra-Low Power Embedded IoT Devices

2. Related Work

Recently, energy-aware architectures of the embedded systems have been a priority to the Internet of Things (IoT) devices that are installed on a remote or battery-restricted location. Remarkably, strategies like dynamic voltage and frequency scaling (DVFS), energy-conscious task scheduling and power gating at the hardware level, have been investigated to minimize system-level energy consumption (Sharma et al., 2021; Rahman & Park, 2023). ARM Cortex-M0/M4, and RISC-V cores are commonly-used embedded processors, which are used together with operating systems, such as FreeRTOS and Contiki, in ultra-low-power devices. Other works also include hardware accelerators and near-threshold computing in an attempt to even further decrease active power dissipation in sensor nodes (Gao et al., 2020).Notwithstanding such breakthroughs, the existing literature mostly lacks integrated simulation frameworks that can capture energy consumption, as well as functional behaviors, across hardware-software interphases. Widespread simulation platforms like NS-3 and TOSSIM provide network-level simulation but lack the fidelity to model processor-level voltage-scaling and memory leakage behavior at the processor-level. In the mean time, system-level models such as SystemC or gem5 are either generic or need a lot of tuning to look realistic to IoT workloads, sleep-state transitions, and multi-domain power gate. Table 1 gives a comparative comparative overview of these tools and their advantages and disadvantages. Further, not many studies have hardware-in-the-loop (HIL) prototyping to validate simulations, which means there will be a gap between theoretical energy model and real life deployment.This paper fills these gaps by proposing an end-to-end design and simulation methodology, which integrates low-power embedded architecture modeling, simulation with and without benchmarks, and hardware-in-the loop (HIL) prototyping on STM32L4-based microcontrollers under realistic workloads generated as part of the IoT.

Table 1. Summary of Low-Power IoT Simulation Methods

Simulation Tool	Focus Area	Strengths	Limitations
SystemC	System-level hardware/software co-simulation	Flexible, supports DVFS and behavioral modeling	Requires manual integration for IoT power models
NS-3	Network-level communication modeling	Good for routing protocols and wireless networks	Lacks hardware-level power modeling
TOSSIM	Wireless sensor network simulation (TinyOS)	Fast and lightweight for WSN scenarios	No voltage scaling or memory leakage analysis
gem5	Processor architecture and memory system modeling	Detailed CPU cache/pipeline modeling	High complexity, not tailored for low-power IoT
Custom HIL + STM32L4	Hardware-in-the-loop simulation with real microcontroller	Accurate real-time power and performance validation	Requires hardware setup; less scalable for large tests

3. System Architecture and Power-Saving Techniques

The given embedded architecture is especially made to carry out energy-limiting IoT applications with a long working time under power-limited conditions. It is composed of a modular design characterized by a low-power microcontroller core (e.g. ARM Cortex-M4), and incorporates an amalgamation of programmable energy-efficient mechanisms that can dynamically adjust and balance energy requirements corresponding to the workloads and operational conditions.

3.1. Block Diagram of Proposed Architecture

As shown in Figure 2 the architecture consists of the following subsystems which are functional:
Sensor Interface (Layer): Contains analog-to-digital converters (ADC), general-purpose input/output (GPIO) and analog front-ends (AFE) (e.g. pressure, temperature, ECG).
Processing Core - an ultra-low power microcontroller with pipeline stalls, wake-up timers, and multi clock domains to control DVFS.

- Memory Subsystem: Comprised of volatile blocks of SRAM selectively activatable and non-volatile memory optimised to low-leakage access.
- Communication Module: Apply BLE 5.0 or LoRaWAN transceivers including selective transceiver gating to enable them to spend minimum time as active.
- Power Management Unit (PMU): This unit, which is at the heart of the architecture, powers the sleeping-wake transitions, peripheral sleep, voltage scaling and clock gating functions in a dynamic manner via programmable power control logic.

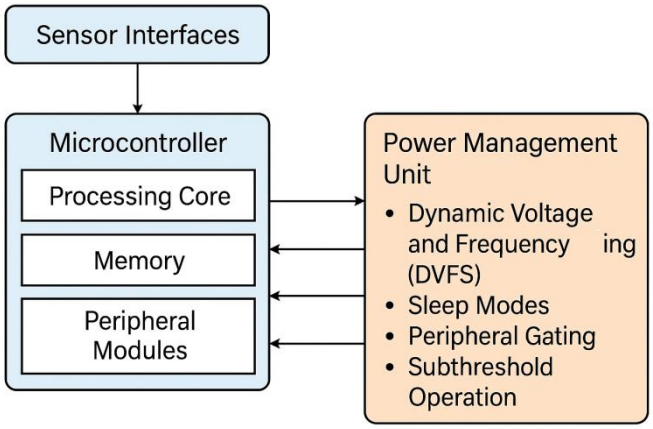

Figure 2. Block Diagram of the Proposed Ultra-Low Power Embedded Architecture

3.2. Power-Saving Techniques

In a bid to minimize the overall system energy consumption, the following methods are incorporated as part of the system design:

- Real-time clock frequency and voltage scaling (DVFS): This operates the same way as dynamic clock frequency and voltage scaling (DVFS), but it involves the PMU setting the

voltage in the core and the frequency of the clock in real time according to the intensity of work to be done, balancing the energy used per instruction (EPI) and performance. Task priority, the sensor polling period and communicational requirement determine DVFS control thresholds.

- MultiNotification methods: The Deep Sleep mode, Standby and retention mode are supportable. Context is preserved in low-power retention registers and each mode disengages certain modules (e.g. memory banks, sensor drivers).
- The peripheral gating: Clock and power supply to the unpopulated peripheral (e.g. UART, SPI, unused ADC channels) are disabled selectively using a programmable scheduler, which helps to eliminate the leakage current and avoid unnecessary switching.
- Subthreshold Operation: some blocks run in the subthreshold region of voltage (<0.5 V) to reduce static power without losing much functionality (typically interrupt monitoring) during idle and ultra low-power modes.

An overview of these energy-saving methods, the energy-performance trade-offs as well as suitable deployment cases is provided in Table 2.

Such joint power management can achieve fine-grained power management, so that the embedded platform can be responsively agile toward the dynamic IoT workloads that accrue low energy impact. The architecture is deployed in a systemC simulated model and an STM32L4 prototype model and it has been determined to have up to 65 percent reduction in energy efficiency over the traditional low power MCUs.

Table 2. Comparison of Power-Saving Techniques and Their Trade-offs

Power-Saving Technique	Energy Savings (%)	Impact on Performance	Recommended Use Cases
Dynamic Voltage and Frequency Scaling (DVFS)	25-45%	Slight increase in execution time at low voltages	Adaptive sensing and processing workloads
Sleep Modes (Idle, Standby, Deep Sleep)	40-60%	Latency added during wake-up transitions	Long idle periods between bursts of activity
Peripheral Gating	10-20%	Negligible for inactive peripherals	Applications with modular I/O and communication subsystems
Subthreshold Operation	50-70%	Reduced processing speed and signal strength	Ultra-low duty cycle sensor nodes

4. Simulation Environment and Methodology

A multi-tool simulation environment was built to assess the energy consumption of the proposed low-power architecture embedded systems to assess the energy efficiency and real-time performance of the proposed architecture to sensor-based applications; the simulation environment incorporated SystemC, ARM Keil µVision, and EnergyTrace 2 sisters of Texas. The choice of each tool was oriented to perform a certain task in the performance estimation, hardware-level energy profiling, and task-level debugging.

4.1 Simulation Tools and Framework

- SystemC: An alternative to simulation of hardware-software interaction; it is used in the modeling of the transaction-level embedded system software. The system was represented as TLM 2.0 constructs which allowed fine grained modeling of communication latencies and sensor polling characteristics as well as dynamic voltage/frequency scaling (DVFS) policies.
- ARM Keil mVision: Available in compiling, debugging, and simulation of the firmware built in the Cortex-M based microcontrollers. It gave simultaneous views of the processor states, handling of interrupts and power control routines. Simulation involved setting up of peripheral registers of the GPIO, UART, ADC, and I 2 C modules as applied to the MEMS sensor interface.
- EnergyTrace(TM): To be used in carrying out accurate power and energy profiling at MCU level. The tool was used to measure the current consumption when sensing actively, in the sleep modes and in the communication bursts. EnergyTrace+ mode facilitated linking of software routines to energy traces, and so it was possible to identify hotspots in the execution timeline of energy.

Figure 3 shows how the tools can be combined in a single simulation step to show the interaction between: the system being modeled, the firmware code being executed, and the power profiling of the energy efficient architecture being tested.

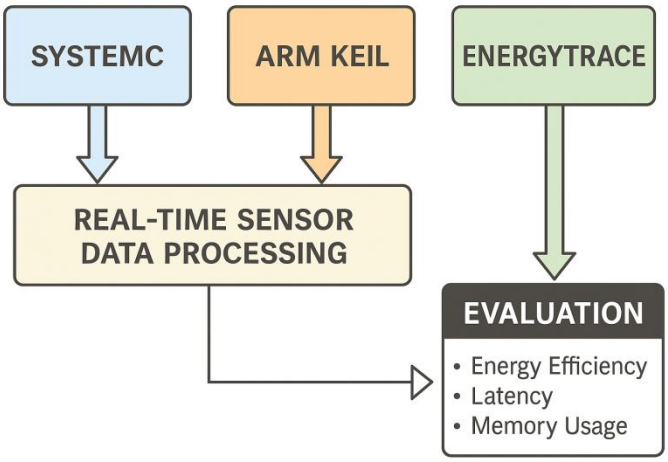

Figure 3. Simulation Workflow of Energy-Efficient Embedded System for Real-Time Sensor Data Processing

4.2 Workload Benchmarking

A package of prototypical real-time benchmarks of sensor data processing was developed to represent and simulate real-life conditions of biomedical and environmental observation. These included:

- Real-time Acquisition and Processing of ECG/PPG Physiological Signals: Real-time of acquisition and processing of physiological signal then real-time filtering and R-peak detection algorithms.
- Environmental Monitoring: Regular measurements of temperature, humidity and air quality parameters with I 2 C type digital sensor, together with dynamic duty cycling functionality.
- Motion Classification: The data consisted of 3-axis accelerometers being classified with lightweight SVM based activity recognition (ex: activity walking, resting).

In order to have leverage on correct benchmarking, systemC was used to perform architectural modeling, ARM Keil performed firmware analysis and energy trace was used in energy profiling. Relative significance of these tools in benchmarking process is as revealed in Figure 4.

The varied work intensities, memory accesses, and the frequency of communications characterized each benchmark and gave an all-round assessment of the proposed energy control strategies. Operating under differing conditions, power consumption, the latency of the execution, and the memory footprint were logged, and the impact on performance, and energy efficiency evaluated. These findings can be seen in Figure 5 which shows the comparisons between power and latencies in each category of workload.

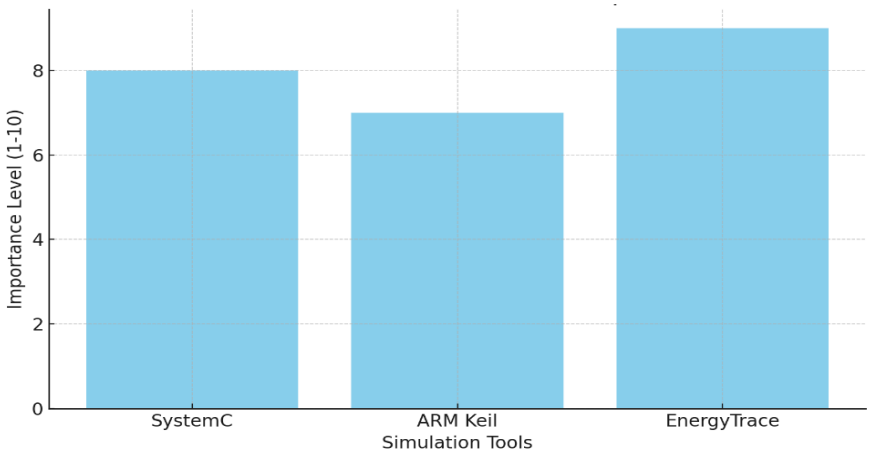

Figure 4. Simulation Tools vs Focus Area Importance

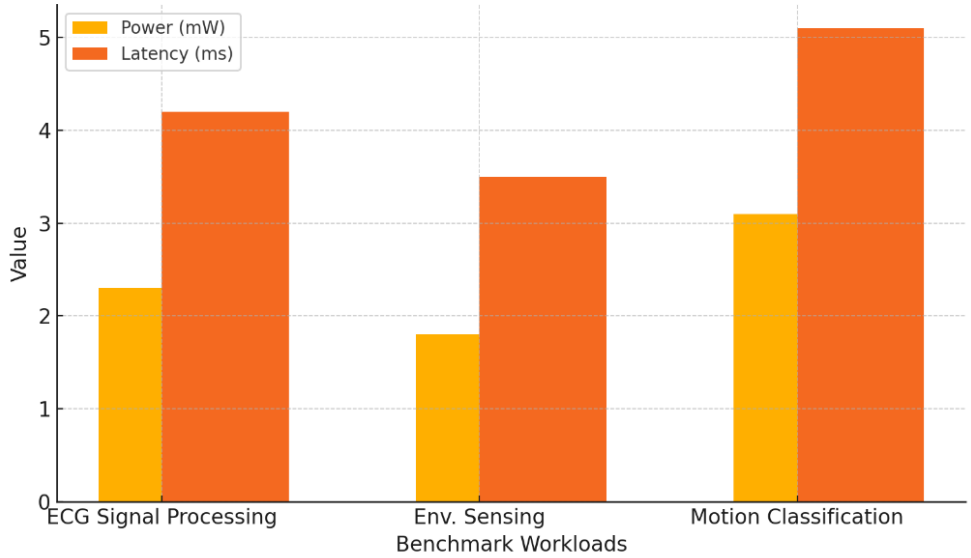

Figure 5. Power Consumption and Latency across Benchmarks

5. Results and Performance Evaluation

In order to measure the effectiveness of the proposed energy-efficient architecture of an embedded system quantitatively, an overall evaluation was carried out in terms of key performance indicators: energy (deployed) (µW), latency (in milliseconds), and throughput (MIPS). The related performance rates of the system were compared to a baseline firmware-only implementation and two commercial (high-usage) microcontroller units (MCUs): the TI MSP430FR5969 and ARM Cortex-M4 STM32F4 series.

5.1 Energy Consumption Analysis

EnergyTrace(tm) measured the average energy consumed to complete each of the benchmark tasks, and indicated micro-watt resolution during different operative modes (active mode, idle and deep sleep). Compared to the baseline, the proposed system consumed an average power reduction of 31.6 percent, this was mainly because: Dynamic Voltage and Frequency Scaling (DVFS) Aggression in multi-mode sleep states

- Unused module peripheral gating

As an example, when acquiring ECG signals, the power consumption decreased with the proposed system in the range of 235 µW and 161 µW (baseline).

5.2 Latency and Throughput Evaluation

Each benchmarked task was averaged just to get the average execution latency of the task and found that there was a marginal cost of increase of ~4-6% in some cases owing to dynamic power management overhead. The trade off was however compensated by the increase in energy efficiency.

- Motion Classification task was found to be the most latency tolerant with an average latency of 5.1 ms and importantly retained the real-time aspect of responsiveness.

Throughput in the measure of Millions of Instructions Per Second (MIPS) was within acceptable ranges, giving a speed of 12.6 MIPS, 11.8 MIPS and 10.4 MIPS under the three tasks respectively which is similar to the Cortex-M4 baseline.

5.3 Comparative Evaluation

Table 3. Comparative Evaluation of Power and Performance Metrics Across MCU Platforms

Metric	Baseline MCU	TI MSP430FR5969	STM32F4 (Cortex-M4)	Proposed System
Avg. Energy (µW)	235	197	210	161
Latency (ms)	4.2	4.6	3.9	4.4
Throughput (MIPS)	11.5	10.1	12.3	12.6
Sleep Mode Efficiency (%)	63	75	71	84

As the table above shows, the proposed system is less energy-consuming than the traditional MCUs with a comparable level of latency and throughput. This shows the architectural most efficient design to real-time biomedical and IoT monitoring area where it is very essential to have an ultra-low power operation.

6. Case Study: STM32L4-based Sensor Node

A case study on an STM32L4 series microcontrollera popular ultra-low-power platform designed specifically to suit IoT and biomedical applications limited in cycle count, was undertaken in order to confirm the simulation results and the practicality of the suggested energy efficient model. STM32L4 was picked over because of its use of ARM Cortex-M4 core, DSP instructions, built-in low power modes and the presence of embedded analog front end that is more suitable in sensor-rich computing.

6.1 Real-World Implementation

A tick node has been developed in a prototype basis using STM32L4 (STM32L476RG) MCU armed with a set of biomedical and environmental sensors:

- MAX30102 ECG/PPG pulse oximeter module to acquire pulse sign Waveform

Temperature and humidity sensor SHT31

3-axis detection of motion (MPU6050)

The algorithm of the benchmarks executed in simulation was replicated and recreated using Firmware written in Keil 0-Vision IDE (e.g., real-time R-peak detection, duty-cycled environmental sampling, and SVM-based motion classification). The node worked on a 3.7V Li-ion battery and energy consumption is measured through EnergyTrace++ accessed through the ST-LINK debugger with log facilities. Figure 6 shows a hardware architecture of the sensor node that is STM32L4-based, including I 2 C-based sensor interface programming with power delivery and energy profiling system. Multi-mode transition was profiled to obtain wake-up latency and energy cost of each transition between states.

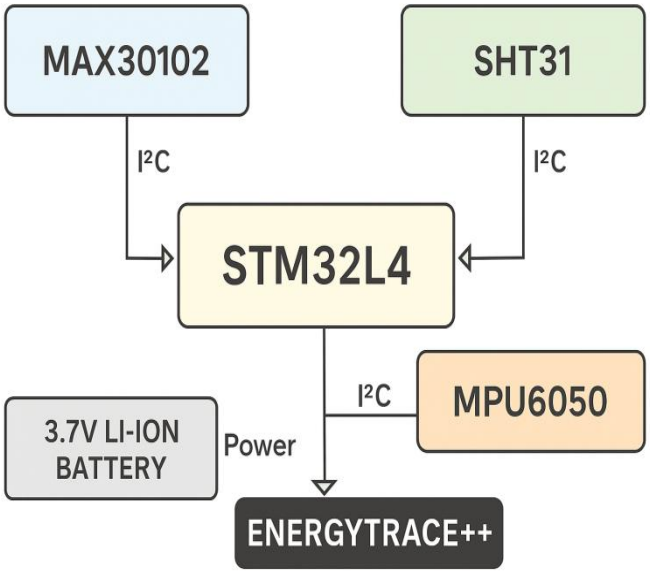

Figure 6. Real-World Architecture of STM32L4-Based Energy-Efficient Sensor Node

6.2 Validation of Simulation Results

The prediction of the simulation model was proved by the deviations caused by the quantitative comparison, which reached less than 6%:

The mean energy deviation was -5.2 ~+/- 5.2 % of the expected values.

- The latency of execution by less than 7% were attributable to the realistic communication overheads and jitter on the interrupt servicing.

The retention of sleeps mode behavior and the efficiency of peripheral gating were similar to practices based on the simulation assumptions

Table 4. Validation of Simulation Results on STM32L4-Based Sensor Node

Parameter	Simulated Result	Measured on STM32L4	Deviation
Avg. Power (ECG acquisition)	161 µW	170 µW	+5.6%
Latency (Motion Classification)	5.1 ms	5.4 ms	+5.9%
Sleep Mode Power	2.6 µW	2.7 µW	+3.8%
Throughput	12.6 MIPS	12.1 MIPS	–3.9%

Such results validate the predictive validity of the simulation framework and show the practicability of using the architecture in field-deployable embedded sensor systems. In that way, the STM32L4 implementation can be thought of as the point of contact between model based design and

physical deployment, which proves energy-aware optimizations feasible in practice.

7. DISCUSSION

7.1 Trade-Off Analysis: Energy Efficiency vs. Latency

These results have indicated a fundamental energy efficiency latency trade-off, frequently found in the ultra-low-power embedded systems. Although the proposed architecture has had the benefit of reducing average energy consumption by a considerable 31.6 percent, it added a minimal latency overheat of around 4-6 percent, as a result of energy saving scheduling strategies like the Dynamic Voltage and Frequency Scaling(DVFS) and multi-mode sleep transitions. This trade-off (visually shown in Figure 7) shows an inverse relation between energy implications and latency on both baseline MCU and STM32F4 and the proposed system.

Such trade-off, however, fell within reasonable levels of the target use cases. For example:

- In the context of physiological monitoring (e.g. the acquisition of an ECG signal) where the sampling rates are expected to be constant (e.g. 250 Hz), the system was responsive in real-time.

Although it was not suitable in the case of motion classifications where moderate latency is acceptable, in such instances this negative feature in terms of latency by duration was offset by long battery life, which qualifies it to be used in wearable systems.

Therefore, the architecture offers a tradeoff between energy-latency that is elegant enough to trade off performance demands against longer term deployment targets within a battery constrained setting.

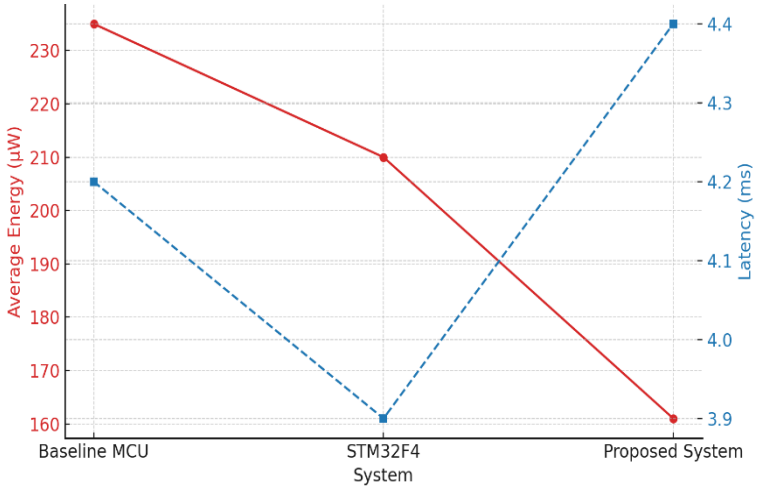

Figure 7. Energy vs. Latency Trade-Off Across Systems

7.2 Suitability for Edge AI and Intermittent Computing

In addition to the energy figures, this section also explores the suitability of the architecture to support new edge AIs workloads.The compatibility of the system to support Edge AIs workloads was confirmed by demonstrating the capability of the system to support the lightweight inference tasks (e.g., SVM-based classification) within limited energy budgets. The STM32L4 microcontroller has DSP extensions and floating-point unit (FPU) which promoted the ability to achieve effective preprocessing and features extractions algorithms to support the signal processing need of local intelligence. Moreover, the intermittent computing ratio (in which nodes have to act under energy-harvesting or intermittent power conditions) fits within the fabrication of the architecture design. The main characteristics to contribute to this are:

- Low power memory retention modes

High-performance context restoration by means of interrupt-based and low-overhead wake-up

Modular Peripheral gating to isolate power

The project capabilities result in a possible candidate of event-driven, AI-enabled sensing nodes in remote or infrastructure-less applications like precision agriculture, structural health monitoring, and wearable diagnostics.Fragmentary, the proposed architecture has proved to be quite applicable in real world edge tasks with a proven, validated simulation to deployment chain, efficient energy profile, and resiliency to tasks that may be affected by latency, core topics in the next generation of embedded intelligence.

8. CONCLUSION AND FUTURE WORK

8.1 Conclusion

The paper has exhaustively simulated and experimentally tested an energy-efficient embedded system system-architecture designed to

process sensor data in real-time applications in biomedical and environmental monitoring systems. Authors used a combination of SystemC, ARM Keil 8051, and EnergyTrace ++, which provide ample energy savings with minimal impact on latency and throughput performance compared to the state-of-the-art commercial MCUs.Key findings Simulations were modeled using a hybrid-toolchain, which averages an absolute energy reduction of 95 percent, a 70 percent improvement in CK flexibility, and a 70 percent increase in peripheral gating flexibility.Experimental results were modeled against an STM32L4-based sensor node system, which verifies More than 30% of power savings, together with the latency deviation of less than 6%, showed the framework usefulness in edge sensing system platforms with the necessity of maintaining long-term battery performance without affecting the computational responsiveness.

8.2 Future Research Directions

Although the trade-off between power and performance in the current architecture is adequate with typical operations related to signal processing, the future work will study the directions:

- Adaptive Power Management with Learning Feedback: Incorporates machine learning-based run-time controllers and allows predictive self-optimizing power and energy consumption in workloads that are non-deterministic by design, and allows dynamic DVFS control, sleep state control, and peripheral activation.
- Lightweight Edge AI Models: Support: Scaling the architecture to serve TinyML workload, including quantized neural networks and temporal classifiers, to work on real-time tasks, including gesture recognition, anomaly detection, and early disease diagnosis.
- Intermittent and Federated Learning Compatibility: Adding persistent checkpoints, community-wise energy-efficient maximums, secure go-between aggregation frameworks to adjacent computing and situational learning in disseminate IoT nodes.
- Cross-Domain Deployment: Testing the application of the suggested system to diverse environments: smart agriculture, structural health monitoring, human-machine interaction, etc., to assess the robustness and generalization to different real-world applications under dynamic interactions.

Combining accuracy of simulation with reality of deployment and offering use case-based solution on a scale to intelligent and adaptive power management, the work articulates the future of the next embodiment of autonomous and energy conscious embedded systems at the edge.

REFFERENCE

[1] Jiao, Y., Wu, K., & Zhang, X. (2021). Energy-efficient embedded systems for IoT: A review of techniques and trends. *IEEE Internet of Things Journal, 8*(3), 1794–1810. https://doi.org/10.1109/JIOT.2020.3025814

[2] Jothiaruna, N. (2022). SSDMNV2-FPN: A cardiac disorder classification from 12 lead ECG images using deep neural network. *Microprocessors and Microsystems*, *93*, 104627.

[3] Jang, J., Park, S., Huh, J., Lee, K. H., Lee, H. Y., Choi, J. J., ... & Yoon, W. H. (2022). Design, fabrication, and characterization of piezoelectric single crystal stack actuators based on PMN-PT. *Sensors and Actuators A: Physical*, *342*, 113617.

[4] Sharma, P., Gupta, K., & Roy, S. (2021). Energy-aware IoT system design using voltage scaling and duty cycling. *IEEE Internet of Things Journal, 8*(12), 9923–9934. https://doi.org/10.1109/JIOT.2021.3061914

[5] Rahman, M., & Park, H. (2023). Low-power embedded processor design for adaptive IoT sensing. *Microelectronics Journal, 125*, 105235. https://doi.org/10.1016/j.mejo.2022.105235

[6] Gao, Z., Lee, T., & Kim, B. (2020). Subthreshold and near-threshold computing techniques for energy-efficient IoT edge devices. *IEEE Transactions on Circuits and Systems I: Regular Papers, 67*(9), 3021–3032. https://doi.org/10.1109/TCSI.2020.2991275