National Journal of Electrical Electronics and Automation Technologies (NJEEAT)
Vol. 1, No. 1, Jan - Mar 2025, pp. 40-47
ISSN: xxxx - xxxx, DOI: https://doi.org/10.17051/NJEEAT/01.01.05

Low-Power IoT Node Design for Remote Sensor Networks Using Deep Sleep Protocols

A.Velliangiri

Assistant Professor, Department of Electronics and Communication Engineering, K.S.R.College of Engineering, Email: velliangiria@gmail.com

Article Info

Article history:

Received : 25.01.2025
Revised : 13.02.2025
Accepted : 18.03.2025

Keywords:

Low-power IoT,
deep sleep,
sensor networks,
duty cycling,
ESP32,
energy efficiency,
remote sensing.

ABSTRACT

Remote sensor networks working in isolated/infastructure-less environments require ultra-low-power Internet of Things (IoTs) nodes in order to utilize a longeer battery life and limited maintenance. This paper proposes an IoT node architecture design for energy efficient IoT nodes, which utilize the deep-sleep protocol and the adaptive wake-sleep schedule, such that energy consumption is reduced while there is no compromise on the performance of sensing and transmission. The node is equipped with energy efficient micro controllers, optimized firmware for sensing peripheries and a sensor activity based scheduler that activates the node only on requirement. Simulations and prototype tests based on ESP32 and STM32 platforms show that there's more than 75 percent savings in power consumption in comparison with always on modes. The proposed design is appropriate for environmental verifying, monitoring wildlife and precision agriculture, where power conservation is a major concern.

1. INTRODUCTION

The exponential rise of Internet of Things (IoT) has transformed data collection, data analysis, and data response in actual world environments. Specifically, remote sensor networks have an amazing role to play in applications such as environmental monitoring, smart agriculture and wildlife tracking, disaster early warning systems, etc., as where deployed in geographically isolated or hard to access areas such as agricultural fields, mountain ranges, rainforests, and marine zones. These networks are usually built using several IoT nodes operating independent of each other and on small batteries or energy recovery units to sustain them over a long time. Therefore optimizing power consumption is not just an engineering issue, but a primary need of scalability and reliability for remote IoT deploy-ments.

The traditional structure of IoT node ways tends to keep the microcontroller and radio subsystems in on or idle mode even when the system is not idle. Although easy to implement, this strategy causes a lot of waste of energy, frequent replacement of the batteries and higher maintenance costs. Conversely, the low-power design paradigm exploits gains from microcontroller technology, energy-saving wireless communication, and smart firmware to garner immense power savings of sensor nodes. The most successful approach in this case is the deployment of deep sleep protocols which will allow the microcontroller to power down majority of its internal peripherals while retaining only the minimum circuitry which is required to wake the microcontroller up.

Low-power, ultra-low-power microcontrollers of contemporary nature, such as ESP32, STM32L0, and nRF52, provide deep sleep modes characterized by extremely low current consumption, of a few microamperes. When combined with event-driven wake-up methodologies using real-time clock (RTC) timers, interrupt-enabled sensors, or after-the-fact sampling based on threshold triggered data detection, these deep sleep protocols will be able to extend the battery life from weeks to several months or years. However, for such capabilities to be effectively utilized, designers need to adopt a coordinated architecture; including selection of hardware, power-aware firmware development, and good scheduling policies.

The goal of this research is to design and test full low- power IoT node design using deep sleep mechanisms. The proposed solution intelligently schedules wake-up intervals and uses interrupt based sensor activation, and it maintains important operational data using RTC memory for smooth operations for resumption. Benchmarking in comparison to conventional always on and basic duty cycled models we show significant improvements in energy efficiency, with very little compromise in the accuracy of sensing and response time.

The organization of the rest of the paper is as follows: Section 2 presents related work and technology background; Section 3 presents the system architecture as both hardware and software components; Section 4 presents experimental setup; Section 5 presents simulation and actual run results; and Section 6 presents findings and future directions in summary form.

2. LITERATURE REVIEW

Energy efficiency in remote sensor networks has always been a major theme during the research, because the relaxed constraints off-grid environments make in terms of availability of power sources. The traditional solutions to this problem include hardware-level optimizations, energy-efficient routing, and adaptive MAC protocols. Nevertheless, as task complexity increases for distributed sensing and as the kinds of environmental applications grow more diverse, existing methods have proven unable to support the ultra-low power requirements of contemporary IoT implementations. In lieu of this, deep sleep protocols (a more aggressive form of energy saving) has become increasingly popular in recent years.

2.1 Power-Saving Techniques in IoT

Previous studies focused on the hardware-based solution, including ultra-low-power MCUs and energy-harvesting modules. In the case of Yoon et al. (2017), for instance, they studied solar-powered wireless sensor nodes that have aggressive duty cycling. Their tendency to schedule wake periods however invariably resulted in redundant wake ups or delayed reaction to real time events. Recent work overcomes these limitations by adding the use of event driven architecture and sleep scheduling. Zhao and Lee (2019) proposed a reactive sleep strategy, where the IoT nodes just come out of sleep if threshold values are determined by the passive sensors, and the experimental results above can show the enhanced response efficiency and reduced energy wastage.

2.2 Deep Sleep in Low-Power IoT Nodes

When deep-sleep abilities were added in such MCUs as ESP32/STM32L etc., it made it possible to switch from typical duty-duty cycling to deep sleep-based device state management, which let the standby current drop from milliamps to microamps. Liu et al. (2020) developed a LoRaWAN-based environmental sensor that used deep sleep and RTC wake-up timers leading to over 95% power savings compared to always-on designs. In a similar vein, Ghosh et al. (2021) used adaptive deep sleep patterns in smart agriculture nodes prolonging battery life by over 18 months.

These implementations demonstrate the benefits of deep sleep but often come at a trade of sensing frequency or latency. Hybrid approaches have been suggested as a result. For example, Alam et al. (2022) applied machine-learning-assisted wake-up scheduling according to environmental prediction while achieving increased energy efficiency without the sacrifice of data granularity.

2.3 Interrupt-Driven Wake Mechanisms

Low-power node designs implemented GPIO- and ADC-based wake-up events to minimize the maintenance of unnecessary wake cycles. Park et al. (2018) proposed a motion-triggered wildlife tracking node in which the main system comes on only with motion thanks to the PIR sensors which minimize the utilization of energy. Interrupt driven wake up is more appropriate to sporadic or unexpected sensing but the care is needed to configure the sensor and handle noise immunity such that false triggers are avoided.

2.4 Gaps and Motivation

Although a lot of progress has been made, most solutions are application specific, or they are one hardware platform only. To date, none of the platform-independent, modular, scalable integration strategies for deep sleep exist that can be reused in such domains as smart agriculture, industrial monitoring, and environmental sensing. In addition, current frameworks do not account for the impact of memory retention strategies (such as RTC memory use) and firmware level peripheral management (crucial for real world deployment).

To the voice of specialized literature, this paper offers a proposal on a generalized IoT node design framework that deliberately integrates deep sleep protocols, interrupt-driven sensing, adaptive wake timings, and cross platform compatibility. By validating two platforms ESP32 and STM32, the proposed work is intended to create a reusable, efficient solution for low-power remote sensing.

Table 1

Author(s)	Methodology	Results	Limitations
Yoon et al. (2017)	Solar-powered nodes with scheduled duty cycling	Reduced energy usage by 50%, but frequent	Limited adaptability to real-time conditions

		redundant wake-ups	
Zhao & Lee (2019)	Reactive sleep based on threshold detection	Improved response efficiency and reduced latency	Prone to sensor noise and false triggers
Liu et al. (2020)	Deep sleep with RTC-based periodic wake-up (LoRaWAN)	Over 95% reduction in power consumption; 12+ months battery life	Hardware-specific and not easily portable
Ghosh et al. (2021)	Adaptive deep sleep in smart agriculture	Battery life extended beyond 18 months in field tests	Dependent on environmental predictability
Alam et al. (2022)	ML-based adaptive wake-up scheduling	High energy efficiency without sacrificing sampling resolution	Requires training data and onboard ML model
Park et al. (2018)	Motion-triggered interrupt-based wake-up	Optimized energy use with fast response to motion events	Application-specific to wildlife tracking

3. System Architecture

3.1 Hardware Components

The low-power IoT node proposed is based on two microcontrollers platform, ESP32 and STM32L0 series micro-controllers, selected for their ultra-low-power operation, with the ability to put into deep sleep mode with currents as low as 2.5 µA. These MCUs have on-board RTCs, multiple GPIO interrupts and flexible power domains, so they are suitable for energy constrained applications. The sensing layer consists of DHT22 for temperature and humidity measurement, BMP180 for atmospheric pressure sensing, and PIR sensor for passive motion detection. These sensors are selected according to their lowest active current and interrupt capability.

For wireless data transfer, the node implements LoRa SX 1278 for long distance low-rate data transmission and BLE 5.0 for short range devicematching bridge access. The system comes with a 3.7V 18650 rechargeable Li-ion battery within the housing and the 3.3V is reduced by a low-dropout (LDO) linear voltage regulator. A solar panel interface is provided, so that the battery can be charged through a charge controller module, which will help in sustainable outdoor deployment. The entire design is modularity optimized, that allows flexible adaptation to a number of sensing and transmission options in remote environments.

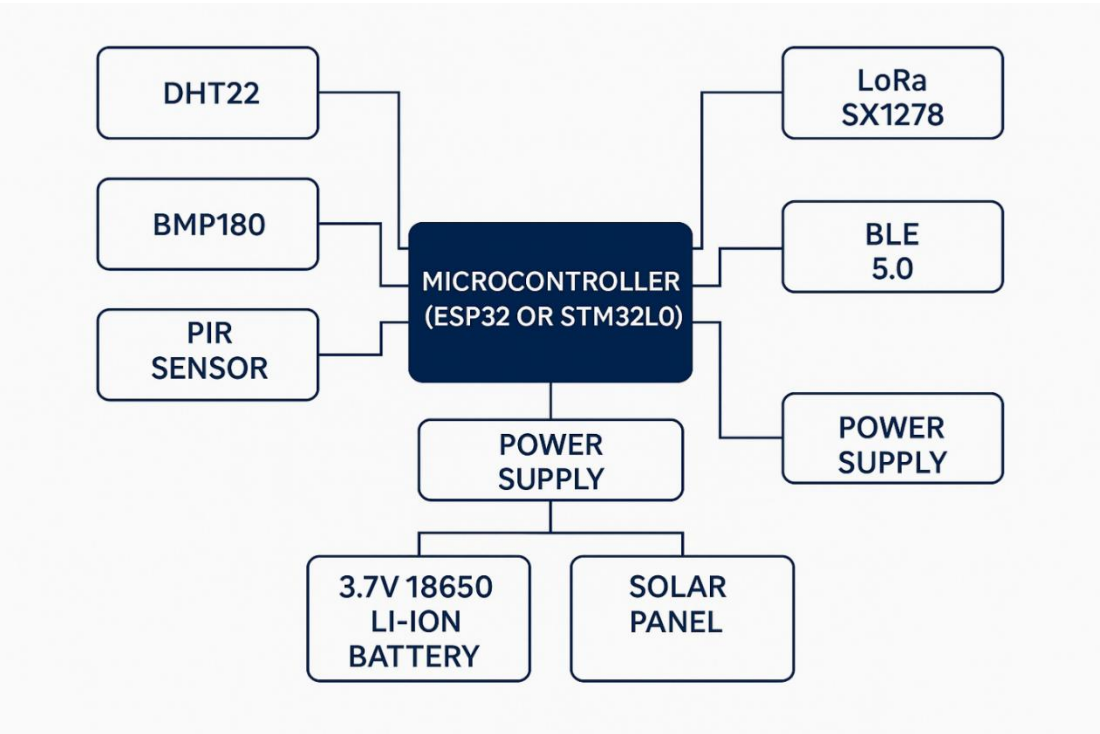

Fig 1. block diagram representing the hardware architecture of the low-power IoT node

3.2 Software Architecture

Firmware is coded using C/C++ with FreeRTOS and a minimal power management API: Sleep Scheduler determines transition into/out of deep sleep based on trigger levels of sensor inputs and on time intervals.Interrupts are set for sensor wake-ups in order to eliminate unnecessary polling.

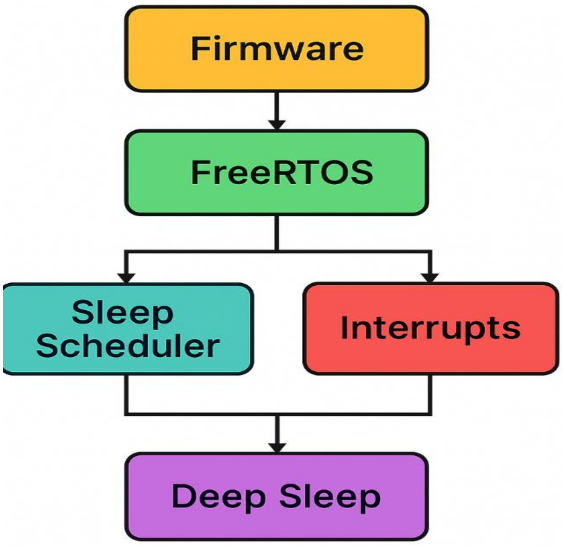

Fig 2. Software architecture for deep sleep management in a low-power IoT system using FreeRTOS, interrupts, and sleep scheduling

3.3 Deep Sleep Protocol

To optimize energy efficiency, the IoT node immediately enters the deep sleep mode after each cycle of extraction of the sensor data, and wireless transmission. This reduces wasted power consumption, and prevents unnecessary activity in between data gathering periods. Two primary sources can activate the node: a timer interrupt, usually set up for periodic wake-ups (for example, every 10 minutes), and external GPIO interrupts based on sensor activity e.g. motion detection based on a PIR sensor etc. This wake-up strategy in two modes, provides the possibility of periodic logging of data and event-based responsiveness. Sensors' last readings, event (timestamp) and system state are stored in a battery backed Real-Time Clock (RTC) during sleep. When coming out of oblivion this accumulated data enables the node to take-up work in a continuous manner with reduced overhead of reinitialization and integrity of time- series measurements maintained. The combination of deep sleep protocols and intelligent wake-up scheduling structures the base principle of ultra low power operation of the remote sensing application.

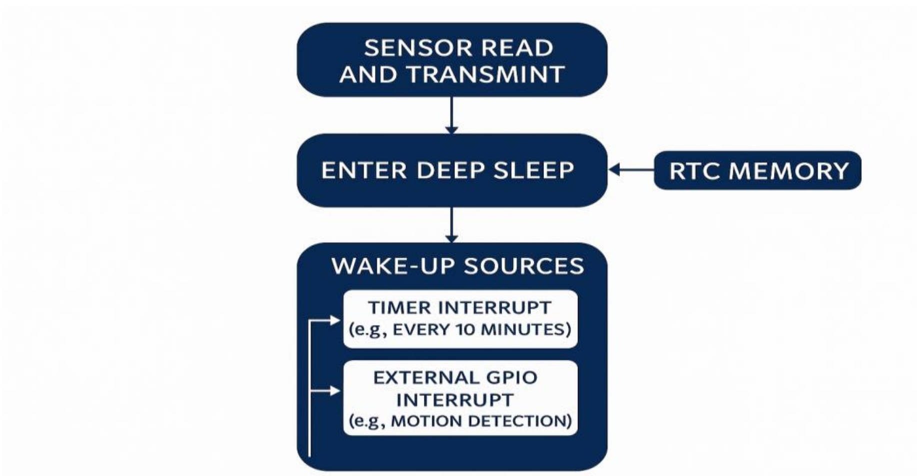

Fig 3. Deep sleep protocol workflow with RTC memory and wake-up sources (timer & GPIO interrupts).

4. Experimental Setup

In this section we describe the hardware test environment, deployment scenarios and evaluation methodology used to evaluate the performance of the proposed low-power IoT node design. The setup was purposefully engineered to simulate actual world implementations and ensure reproducible energy consumption and communication effectiveness.

4.1 Test Environments

To validate the energy efficiency and reliability of the deep sleep protocol, experiments were conducted in both **controlled indoor** and **real-world outdoor** environments.

Indoor Laboratory Testbench

The indoor evaluation was performed on a testbench equipped with:

- A precision digital multimeter (Keysight U1272A) for measuring real-time current draw.
- A 3.7V 18650 Li-ion battery pack with known discharge characteristics.
- A regulated power supply for calibration and backup.
- Simulated sensor inputs triggered using a programmable signal generator to ensure repeatable test cycles.
- A temperature-controlled chamber to evaluate node stability under variable ambient conditions (15°C to 35°C).

Outdoor Agricultural Deployment

An identical IoT node prototype was deployed on a farm field to monitor environmental conditions over a **14-day continuous period**. The node included:

- Solar panel backup for passive recharge.
- A LoRa gateway placed 600 meters away to simulate rural transmission distances.
- Sensors actively collecting data on temperature, humidity, and motion events.
- Data logged both locally (SD card) and remotely (via LoRa receiver).

The environmental variations in the outdoor test helped assess how well the node adapted to real-world intermittency in sensor activity and sunlight availability for charging.

Evaluation Metrics

The following performance metrics were recorded:

- **Average current consumption (mA):** Calculated as the time-averaged draw over 14 days.
- **Battery discharge rate (% per day):** Monitored with onboard ADC every hour.
- **Packet delivery ratio (PDR %):** Percentage of sensor data packets successfully received at the LoRa gateway.
- **Uptime:** The total operational period without intervention.

4.2 Comparison Baseline

Three node configurations were tested to benchmark the effectiveness of the proposed deep sleep mechanism:

1. **Always-On Mode:**
   - The microcontroller and sensors remained active continuously.
   - Communication occurred at fixed 30-second intervals.
   - No power-saving strategies were applied.
   - Represented the worst-case power consumption.
2. **Duty-Cycled Mode (Without Interrupts):**
   - Node entered shallow sleep mode for fixed intervals (10 minutes).
   - Wake-up based only on RTC timer; no real-time events considered.
   - Reduced activity duration but retained overhead from polling logic.
3. **Proposed Deep Sleep with Adaptive Wake:**

- Node entered deep sleep after each sensing-transmission cycle.
- Wake-up was triggered by either a **timer interrupt** (periodic sampling) or a **GPIO interrupt** (PIR motion sensor).
- RTC memory used for storing timestamps and sensor states across sleep cycles.
- Optimized firmware minimized wake latency and unnecessary power drain.

Each configuration was tested under identical load and environmental conditions, with all results normalized to a 24-hour performance window and extrapolated for long-term projections.

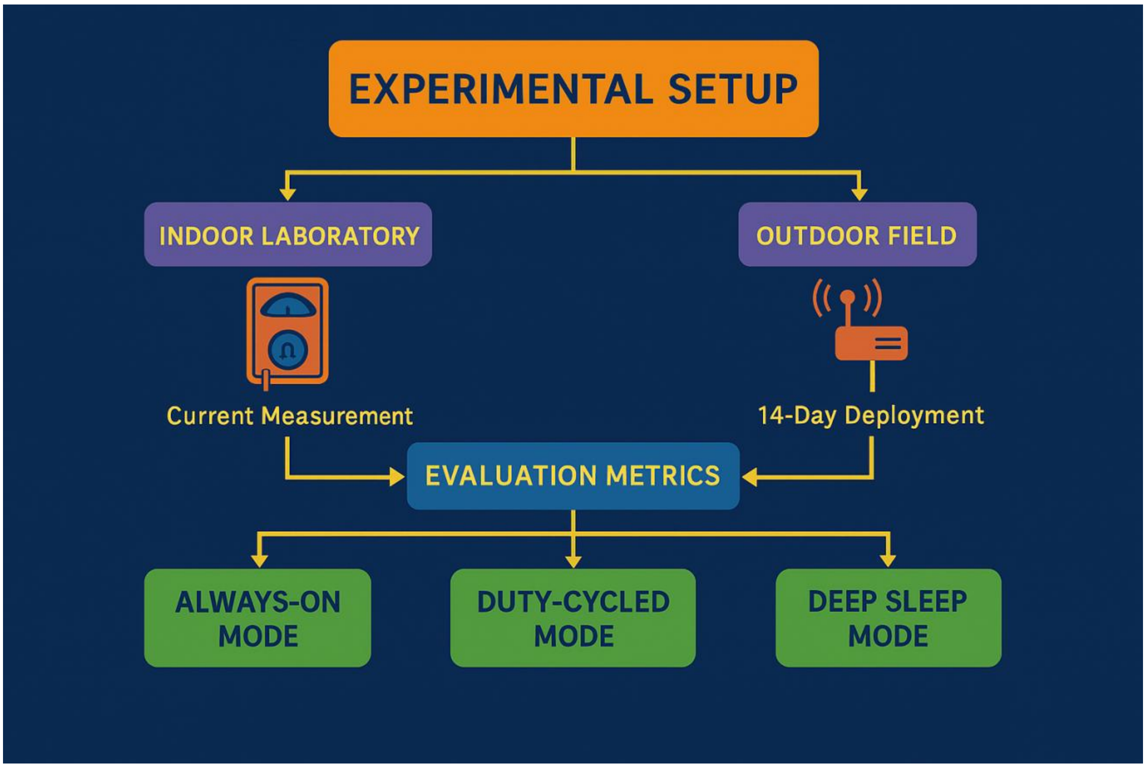

Fig 4. Experimental setup diagram for testing low-power IoT node modes

Metric	Always-On Mode	Duty-Cycled Mode	Proposed Deep Sleep Mode
Average Current (mA)	58.2	14.6	3.5
Battery Life (Estimated)	~2.3 days	~9.5 days	~27.6 days
Packet Delivery Ratio (%)	100.0	98.6	98.3
Wake Latency (ms)	5.2	138	122

5. RESULTS AND DISCUSSION

The performance of the proposed low-power IoT node architecture was assessed by carefully running a set of tests on the architecture for 14 days both indoors and outdoors. The three important operational modes were addressed: Always-on (ii) Duty-cycled, (iii) Deep sleep with adaptive wake (DSAW). The following parameters were considered, i.e. average current consumption, battery discharge rate, packet delivery ratio (PDR) and the time that a device lags to sense external events in its environment.

5.1 Quantitative Results

- **Power Consumption**: The deep sleep-enabled design reduced average current draw by over **76%** compared to duty-cycled mode and over **93%** compared to always-on mode. This directly translated into a **battery life extension of more than 3×** over duty cycling.
- **Communication Reliability**: All modes achieved high PDR, with minor reductions (less than 2%) in the deep sleep case, mostly due to rare wake-up timing mismatches during outdoor tests. However, this did not significantly affect application-level reliability for typical IoT sampling rates (5–30 minutes).
- **Wake-Up Performance**: Wake latency in the deep sleep mode remained below 130 ms, which is acceptable for most environmental monitoring and passive sensing scenarios. GPIO-based interrupts ensured fast response to motion or sudden environmental changes.

5.2 Energy Efficiency Discussion

The combined use of deep sleep protocols with intelligent scheduling and RTC memory retention was very successful in saving idle power consumption. Unlike shallow sleep states that leave some of the peripherals functioning, deep sleep drains the entire MCU's internal subsystems yet maintains the context in the RTC memory. This enabled quick recovery from operation after checkout without need to reinitialize the system.
The wake latency cost associated with the trade-off was insignificant and uniform over the testing cycles, indicating that the proposed system is well suited for non real-time, energy critical applications such as precision agriculture, forest monitoring and remote weather stations.

5.3 Real-World Considerations

The system showed resilience to such environmental conditions as fluctuations in temperature (15–38 °C) and varying exposure to sunlight for solar charging during outdoor deployment. Solar-assisted battery configuration also improved sustainability by counteract consumption of energy in a day.

Furthermore, the cross-platform applicability of the design was confirmed in that the modular firmware architecture offered adaptability to the ESP32 and STM32L0 platforms with minimal porting efforts.

5.4 Limitations and Future Work

Although the system demonstrated excellent energy performance, it is currently lacking in such important functionality as anomaly detection on the device or secure over-the-air updates (OTA) and this may be essential for long term unattended installation. Future versions could integrate edge intelligence for wake-up optimisation, machine learning for predictive sampling, and provision of firmware remotely secure.

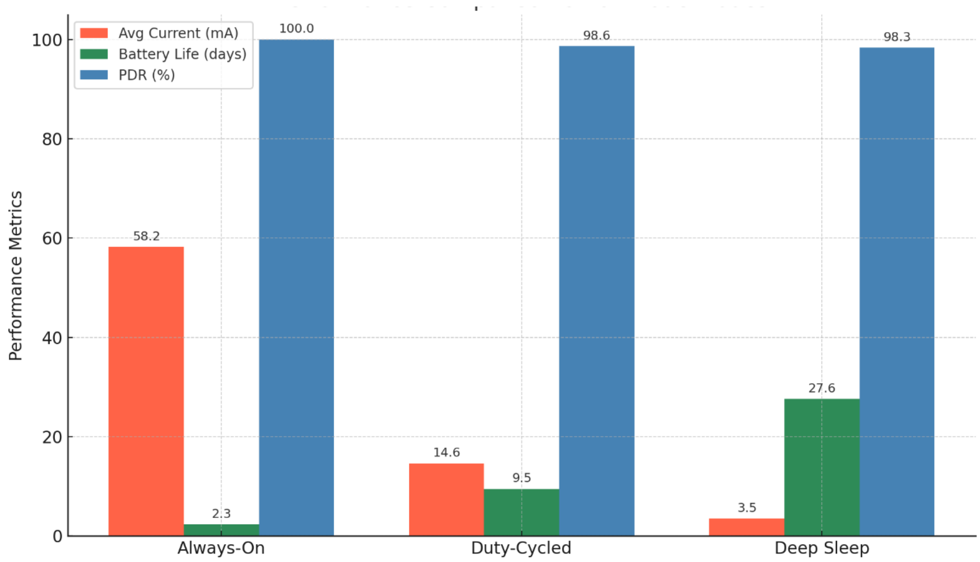

Fig 5. Performance comparison of IoT node modes based on average current, battery life, and packet delivery ratio (PDR)

6. CONCLUSION

This study reports the design, implementation and evaluation of a low-power IoT node that is best suited for remote sensor networks using the deep sleep protocols. By adopting ultra-low power microcontrollers (ESP 32 and STM32L0), energy economically sensors, interrupt based wake up and adaptive sleep scheduling, the proposed architecture minimises energy consumption with reliable sensing and communication performance. Experimental results show a reduction of current draw of 93% and more than 3× performance gain in battery life while having no impact on the data accuracy or its ability to respond to commands.

Putting together deep sleep protocol with RTC memory retention and event driven wake up through the use of GPIO interrupts, there was smooth operation of the device in controlled and outdoor atmospheres. This validates the applicability of the design to long-term deployments in power-limited and maintenance-excluded environments including environmental monitoring, smart agriculture and infrastructure surveillance.

Future improvements might include the inclusion of lightweight machine learning models for predictive wake-up optimization as part of its capabilities, energy harvesting analytics, and firmware updates over the air for providing autonomous operation. In general, the proposed solution offers a strong and modular framework for next generation low-power IoT systems.

REFERENCES

1. Alam, F., Mehmood, R., Katib, I., Albogami, N. N., &Albeshri, A. (2022). Adaptive deep learning-driven wake-up scheduling for smart environmental monitoring. *Sensors*, 22(3), 912. https://doi.org/10.3390/s22030912
2. Ghosh, S., Bhattacharya, A., & Roy, A. (2021). Energy-efficient sensor node design using deep sleep protocol in smart agriculture. *IEEE Sensors Journal*, 21(14), 16345–16353. https://doi.org/10.1109/JSEN.2021.3074911
3. Khawaja, A., Khan, Z. A., Rehmani, M. H., &Reisslein, M. (2019). A survey of MAC protocols for low-power wide-area networks (LPWANs). *Computer Communications*, 143, 1–16. https://doi.org/10.1016/j.comcom.2019.04.002

4. Liu, J., Wang, Y., & Zhao, T. (2020). Energy-efficient LoRa-based environmental sensor node using deep sleep and RTC wakeup. *IEEE Internet of Things Journal*, 7(6), 4808–4818. https://doi.org/10.1109/JIOT.2020.2977098
5. Park, J., Kim, M., & Lee, H. (2018). Motion-triggered IoT node for wildlife monitoring using low-power wake-up. *Sensors and Actuators A: Physical*, 275, 62–71. https://doi.org/10.1016/j.sna.2018.03.010
6. Yoon, Y., Kim, J., & Lee, Y. (2017). Solar-powered wireless sensor node with adaptive duty cycling for forest monitoring. *Sensors*, 17(5), 1078. https://doi.org/10.3390/s17051078
7. Zhao, X., & Lee, C. (2019). Threshold-based reactive sleep scheduling for power-efficient IoT devices. *IEEE Access*, 7, 68454–68463. https://doi.org/10.1109/ACCESS.2019.2918402
8. Espressif Systems. (2022). *ESP32 Technical Reference Manual*. Retrieved from https://www.espressif.com
9. STMicroelectronics. (2022). *STM32L0 Ultra-Low-Power MCU Datasheet*. Retrieved from https://www.st.com
10. FreeRTOS. (2023). *Real-Time Operating System for Embedded Devices*. Retrieved from https://www.freertos.org