

Energy-Efficient IoT Sensor Design: A Comprehensive Review of Sensor Technologies, Power Consumption, and Sustainable Deployment

Dr. V.Valarmathi¹, Dr.S.Nandhini²

^{1,2} Associate Professor, Department of IT & CG, School of Computing Systems, Sri Krishna Knowledge Foundation Deemed to be university, Coimbatore, Tamil Nadu, India.

Article Info

Article History:

Published: 16 Sept 2026

Publication Issue:

Volume 3, Issue 9
September-2026

Page Number:

455-472

Corresponding Author:

Dr. V.Valarmathi

Abstract:

Internet of Things (IoT) systems depend on distributed sensors to observe physical, environmental, and operational conditions and convert those observations into actionable digital information. Although sensing capability has expanded rapidly, energy availability remains a major constraint for battery-powered and remotely deployed devices. This review examines IoT sensor technologies from an energy-efficiency perspective, linking sensor characteristics, measurement requirements, processing, wireless communication, power-management strategies, and environmental integration. The paper consolidates a practical taxonomy of temperature, humidity, pressure, motion, proximity, light, acceleration, positioning, gas, and sound sensors and discusses representative devices and indicative energy requirements. It further examines the main sources of energy consumption, including sensing, computation, communication, idle operation, environmental effects, and power-conversion overhead. A structured framework is proposed for energy-efficient sensor design based on application requirements, low-power hardware, adaptive sensing, local processing, communication optimization, energy harvesting, and continuous power profiling. Special attention is given to duty cycling, adaptive sampling, data aggregation, compression, edge computing, low-power wireless technologies, and energy harvesting. The review also discusses the relationship between energy consumption, battery lifetime, operational expenditure, reliability, scalability, and sustainability. Finally, challenges and emerging directions including context-aware sensing, machine-learning-assisted energy management, secure low-power operation, and sustainable environmental monitoring are identified. The paper is intended to support researchers, system designers, and practitioners in selecting sensors and designing IoT architectures that balance measurement quality, responsiveness, network lifetime, and energy sustainability.

Keywords: Internet of Things, IoT sensors, energy efficiency, sensor networks, duty cycling, adaptive sensing, energy harvesting, edge computing, low-power communication, sustainable IoT

1. Introduction

IoT has evolved into a distributed computing paradigm in which physical objects are equipped with sensing, communication, processing, and actuation capabilities. Sensors form the observation layer of this paradigm because they transform physical conditions into measurable signals that can be stored, transmitted, analyzed, and used to trigger actions. The source material for this paper emphasizes that IoT sensors collect information such as temperature, humidity, light, motion, pressure, and sound and connect the physical environment with digital systems. This role supports real-time monitoring, remote sensing, automation, contextual awareness, analytics, and environmental management. The growing scale of IoT deployments, however, changes the design problem: a sensor is not evaluated

only by whether it can measure a variable, but also by how efficiently it performs that task over its intended lifetime. Energy efficiency therefore becomes a system-level requirement rather than a secondary hardware characteristic.

Energy constraints are particularly important when sensors are battery-powered, mobile, installed in difficult-to-access locations, or deployed in large numbers. A sensor that consumes excessive energy may require frequent maintenance, battery replacement, or recharging, thereby increasing operational cost and potentially interrupting data collection. The supplied material identifies prolonged battery life, cost savings, sustainability, scalability, reliability, mobility, energy harvesting, and network capacity as major reasons for prioritizing energy-efficient design. These considerations are consistent with the broader IoT literature, which treats energy availability as a fundamental limitation in sensor networks and Green-IoT systems. Reviews of energy efficiency show that sensing, processing, communication, duty cycling, and network organization must be considered together rather than optimized independently. [1]–[5] The result is a multi-objective design problem in which energy savings must be balanced against accuracy, latency, reliability, coverage, and application requirements.

The purpose of this review is to organize the practical information supplied about IoT sensors into a journal-style framework and extend it into a coherent discussion of energy-aware design. The paper first establishes a taxonomy of common sensors and representative devices. It then examines energy-consumption mechanisms and methods for profiling power usage. Next, it discusses hardware selection, adaptive sampling, sleep and wake-up control, communication optimization, data reduction, edge processing, and energy harvesting. Environmental monitoring integration is considered because it demonstrates how heterogeneous sensors can provide richer information while creating additional energy and interoperability requirements. Finally, current limitations and emerging research directions are discussed, followed by a proposed evaluation framework for comparing energy-efficient IoT sensor designs.

An important distinction is made between sensor energy consumption and complete node energy consumption. A sensing element may require only a small current while an attached microcontroller, voltage regulator, memory device, wireless radio, indicator, or actuator dominates the energy budget. Similarly, a nominally low-power communication protocol may consume substantial energy if the application transmits too frequently, uses unnecessarily large packets, or repeatedly reconnects. Therefore, energy efficiency should be measured at the system level and under realistic operating states. The source material highlights sensing, data processing, transmission, communication protocol choice, power management, hardware selection, energy harvesting, and environmental conditions as interacting contributors. This systems view is essential for avoiding misleading comparisons based only on a component's nominal active current.

The significance of energy-aware IoT design extends beyond battery lifetime. Large deployments can contain hundreds or thousands of nodes, so even small per-node savings can produce substantial aggregate benefits. Lower energy demand can reduce maintenance visits, extend deployment periods, enable denser networks, and make renewable or harvested power more practical. Energy efficiency can also support environmental objectives by reducing wasted energy and the associated carbon footprint of power generation. At the same time, aggressive power reduction can reduce sampling frequency, processing capability, or communication availability, which may degrade data quality. Consequently, an energy-efficient architecture should not simply minimize power; it should minimize unnecessary energy expenditure while preserving the level of service required by the application. This principle provides the foundation for the sensor taxonomy, analysis, and design framework developed in the following sections.

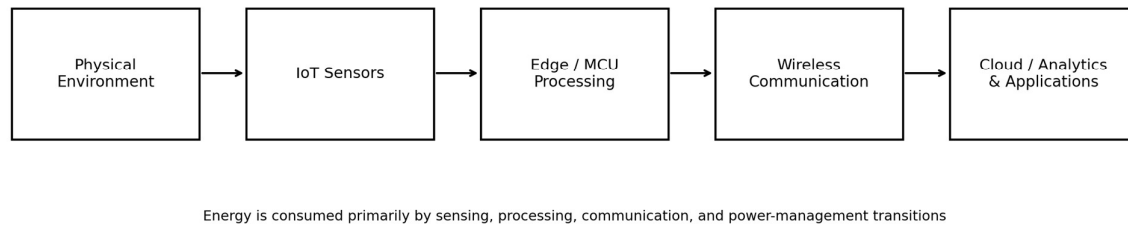


Figure 1. Generic IoT sensor-to-analytics data path and major energy-consuming stages.

IoT sensor design also needs to account for the relationship between physical sensing and digital representation. Every measurement introduces uncertainty through sensor noise, calibration error, quantization, drift, and environmental interference. Increasing sampling frequency does not automatically improve information quality if the underlying signal changes slowly or the sensor accuracy is the limiting factor. Energy-aware design can therefore use signal characteristics to determine the minimum useful sampling rate. Statistical summaries, threshold detection, and change-point detection can reduce unnecessary acquisition and communication. This is especially useful in environmental applications where many consecutive measurements may be highly correlated. By treating information value as a design variable, the sensor node can spend energy selectively rather than continuously operating at its maximum capability.

Another important consideration is deployment scale. A prototype may contain only one sensor node, but a practical IoT system can include hundreds or thousands of nodes distributed over buildings, farms, factories, roads, or environmental regions. A small inefficiency repeated across many devices can become a substantial system-level burden. Conversely, a modest reduction in average energy per node can extend network lifetime and reduce maintenance requirements. Scalability also changes the importance of communication overhead, synchronization, firmware updates, and fault management. Energy-efficient design should therefore be evaluated at both node and network levels. This review uses the sensor node as the fundamental unit while recognizing that sustainable IoT performance ultimately depends on the behavior of the complete distributed system.

2. Taxonomy and Characteristics of IoT Sensors

IoT sensors can be classified according to the physical quantity they measure, their sensing principle, their output interface, and their deployment context. Common environmental and physical sensors include temperature, humidity, pressure, motion, proximity, light, acceleration, positioning, gas, and sound sensors. Temperature sensors may be implemented using thermocouples, thermistors, or resistance temperature detectors (RTDs), while humidity sensors can use capacitive, resistive, or gravimetric principles. Pressure sensing may rely on piezoresistive, capacitive, or strain-gauge structures. Motion sensing commonly uses passive infrared, ultrasonic, or microwave techniques. This diversity is valuable because IoT systems can select a sensor that matches the required measurement range, precision, response time, physical size, and power budget. However, sensor selection should also consider standby current, active current, warm-up time, communication interface, calibration needs, environmental tolerance, and the frequency with which measurements must be obtained.

Temperature sensing is among the most widely used IoT functions because temperature is a key contextual variable in buildings, industrial processes, logistics, agriculture, healthcare equipment, and cold-chain systems. The supplied

material identifies thermocouples, thermistors, and RTDs as representative sensor technologies and degrees Celsius or Fahrenheit as measurement units. Humidity sensors complement temperature sensors in greenhouses, indoor farming, HVAC systems, and manufacturing environments. Pressure sensors are used in automotive monitoring, industrial automation, and healthcare. Motion and proximity sensors are especially useful when event-driven operation is possible because the node can remain asleep until movement or object presence is detected. Light sensors measure illuminance, commonly in lux, and can support adaptive lighting and energy management. In each case, the application determines the acceptable trade-off between measurement frequency and energy consumption.

Accelerometers, positioning modules, gas sensors, and sound sensors extend the sensing layer into mobility, safety, environmental intelligence, and human interaction. Accelerometers measure acceleration in units such as metres per second squared and are widely used in wearables, vehicle monitoring, and motion-controlled systems. GPS or GNSS modules provide latitude, longitude, and altitude information but can require considerably more energy than simple environmental sensors because satellite acquisition and tracking involve active radio-frequency processing. Gas sensors can detect concentrations in ppm or percentage units and are relevant to air-quality monitoring and industrial safety. Sound sensors, including microphones and MEMS microphones, convert acoustic signals into electrical representations and can support voice interaction, noise monitoring, and event detection. These examples demonstrate why a single energy-efficiency strategy cannot be applied uniformly across all sensors.

Table 1 consolidates the major sensor categories supplied in the source material and maps them to representative sensing technologies, units, uses, and applications. The table is intended as a design-oriented taxonomy rather than an exhaustive catalogue. Actual sensor specifications vary by manufacturer, operating mode, sampling rate, interface, environmental condition, and firmware configuration. Consequently, researchers preparing experimental work should report the exact part number, supply voltage, sampling configuration, measurement resolution, and active and sleep currents. This is particularly important when comparing energy performance because two devices described by the same broad category can exhibit very different energy profiles. A publishable experimental paper should therefore distinguish between nominal datasheet characteristics and measured system-level consumption.

Representative models such as DHT22, SHT31, BMP280, HC-SR501, E18-D80NK, TSL2561, MPU-6050, NEO-6M, MQ-135, and MAX9814 are useful examples for prototyping and educational comparison. The supplied source associates these devices with different sensor types and indicative power requirements. These values should be treated as approximate design references rather than universal specifications, because operating modes and module implementations can materially change consumption. In particular, a complete breakout board can consume more power than the underlying integrated sensor because of regulators, LEDs, pull-up resistors, level shifters, or interface circuitry. For journal publication, the most defensible approach is to cite the manufacturer datasheet for the exact revision and report independently measured current and energy where possible.

Sensor	Representative sensor name	Units	Uses	Applications
Temperature	Thermocouple, Thermistor, RTD	°C, °F	Climate control, industrial processes, food storage	HVAC, smart homes, cold-chain monitoring
Humidity	Capacitive, Resistive, Gravimetric	% RH	Agriculture, HVAC, manufacturing	Greenhouses, indoor farming, moisture control
Pressure	Piezoresistive, Capacitive, Strain gauge	Pa, bar, PSI	Tire monitoring, automation, healthcare	Automotive, healthcare, industrial processes

Motion	PIR, Ultrasonic, Microwave	Event / motion state	Security, lighting, occupancy	Home security, smart lighting, buildings
Proximity	Inductive, Capacitive, Ultrasonic	Presence distance /	Object detection, parking, touchless control	Parking guidance, industrial sensing
Light	Photodiode, Phototransistor, ALS	lux	Lighting, display adjustment, energy saving	Smart lighting, displays, energy management
Accelerometer	MEMS, Piezoelectric, Capacitive	m/s ²	Fitness, vehicles, motion control	Wearables, vehicle tracking, controllers
GPS/GNSS	Receiver, GNSS module, tracker	Latitude, longitude, altitude	Navigation, asset tracking	Fleet management, outdoor navigation
Gas	Electrochemical, Photoionization, Infrared	ppm, %	Air quality, safety	Indoor air quality, leak detection
Sound	Microphone, MEMS microphone, electret condenser	dB	Speech, noise monitoring, automation	Voice assistants, sound analysis, smart audio

Table 1. IoT sensor categories, sensing technologies, units, uses, and applications.

Category	Model	Type	Indicative energy/power characteristic
Temperature	DHT22	Digital temperature and humidity	~2.5 mA active
Humidity	SHT31	Digital humidity and temperature	~2.7 μ A in a low-power operating condition (verify mode)
Pressure	BMP280	Barometric pressure	~2.7 μ A in low-power operation (verify mode)
Motion	HC-SR501	PIR motion	~50–60 μ A (module-dependent)
Proximity	E18-D80NK	Photoelectric proximity	~15–20 mA (module-dependent)
Light	TSL2561	Digital light	~0.7 μ A in low-power operation (verify mode)
Accelerometer	MPU-6050	MEMS accelerometer + gyroscope	~3.9 mA active (configuration-dependent)

GPS	NEO-6M	GPS module	~45 mA active (module-dependent)
Gas	MQ-135	Air-quality gas sensor	~150–200 mW typical heater-related range (module-dependent)
Sound	MAX9814	Electret microphone amplifier	~0.5 mA (configuration-dependent)

Table 2. Representative sensor models and indicative consumption.

Sensor selection should also consider interface and integration characteristics. Analog sensors may require an analog-to-digital converter and signal-conditioning circuitry, while digital sensors can provide calibrated measurements through interfaces such as I2C, SPI, UART, or pulse-based outputs. The interface affects not only development complexity but also energy consumption because communication between the sensor and microcontroller requires clocking and data transfer. Some sensors provide internal filtering, calibration, compensation, or measurement storage, which can reduce external processing requirements. Others require continuous excitation or periodic warm-up. Therefore, the most energy-efficient sensor is not necessarily the device with the lowest datasheet current; it is the device that provides the required information with the lowest total system energy.

Sensor characteristics should be documented using a common set of parameters when preparing a journal comparison. Recommended fields include measurement range, accuracy, resolution, response time, supply voltage, active current, standby current, warm-up time, communication interface, physical dimensions, environmental rating, and estimated cost. For wireless or location modules, communication conditions and acquisition time should also be recorded. Such a structured description improves reproducibility and makes it easier for readers to understand why one sensor was selected over another. The tables in this paper therefore serve as a starting taxonomy, while detailed experimental studies should expand them with verified manufacturer specifications and measured performance.

3. Energy Consumption Mechanisms in IoT Sensor Nodes

Energy consumption in an IoT sensor node is the result of several operating states rather than a single constant value. A practical node can alternate among sensing, processing, memory access, radio transmission, radio reception, idle, and deep-sleep states. The average energy requirement depends on the duration and frequency of each state. A useful conceptual expression is

$$E_{\text{total}} = E_{\text{sense}} + E_{\text{process}} + E_{\text{tx}} + E_{\text{rx}} + E_{\text{idle}} + E_{\text{sleep}} + E_{\text{aux}},$$

where the auxiliary term includes regulators, indicators, clock sources, and other support circuits. This formulation makes clear why a low-power sensor element does not automatically produce a low-power IoT node. If the radio remains active for long periods, or if a microcontroller repeatedly wakes for small tasks, communication and state-transition energy may dominate the sensor itself.

Sensing and measurement consume energy according to the sensing principle, excitation requirements, measurement resolution, conversion process, and sampling frequency. Some sensors require an internal heater, optical source, bias circuit, or excitation voltage, whereas others can remain in an ultra-low-power standby mode. Increasing the sampling rate usually increases energy because the sensing element, analog front end, analog-to-digital conversion, and processing pipeline are activated more frequently. Higher resolution can also increase processing and memory requirements. Therefore, the application should specify the minimum temporal and numerical resolution required rather than maximizing these parameters by default. Event-driven sensing is especially effective for applications in which meaningful changes are relatively infrequent. A motion detector, for example, may consume less energy when it detects a threshold crossing and wakes a higher-performance subsystem only when required.

Processing energy is influenced by the microcontroller architecture, clock frequency, memory operations, algorithm complexity, and amount of local computation. Simple thresholding can generally be performed with less computational effort than feature extraction or machine-learning inference. However, local processing can reduce communication energy by filtering or aggregating data before transmission. The energy-optimal choice therefore depends on the relative cost of computation and communication. Edge computing can be beneficial when a small amount of local computation eliminates large volumes of redundant wireless traffic. The source material identifies algorithm optimization, reduced unnecessary computation, and energy-efficient processing as important measures. In a rigorous study, researchers should measure both computation time and energy rather than assuming that local processing is always more efficient.

Communication is frequently a major component of energy consumption because wireless radios require power for startup, channel access, transmission, reception, acknowledgements, and retransmissions. The source material highlights Wi-Fi and cellular communication as generally more energy demanding than low-power technologies designed for IoT, while Zigbee, Z-Wave, BLE, and LoRaWAN are presented as energy-oriented alternatives. The correct technology nevertheless depends on distance, payload size, latency, network density, reliability, spectrum, and application constraints. Repeated transmissions caused by poor link quality can eliminate the theoretical benefit of a low-power protocol. Similarly, long-range transmission can require more energy per message even if the device transmits infrequently. Energy analysis should therefore consider both radio power and the number and duration of communication events.

Environmental conditions also affect energy requirements. Temperature and humidity can influence electronics, battery characteristics, sensor accuracy, and the stability of harvested energy. Optical sensors may behave differently under changing illumination, while outdoor nodes may face weather conditions that affect enclosure requirements and communication reliability. A robust energy model should therefore include environmental context rather than assuming constant conditions. The supplied material specifically identifies environmental variability as a contributor to energy consumption and a challenge for optimization. In addition, power-management decisions can influence data quality. Excessive sleep periods may cause missed events, while overly aggressive sampling reduction may hide important trends. Energy efficiency must therefore be defined as useful information obtained per unit of energy, not merely minimum electrical power.

Energy consumption can also be understood through the distinction between static and dynamic demand. Static or background demand occurs when a component remains powered even though it is not actively producing data. Dynamic demand occurs when sensing, computation, memory access, or communication is performed. Reducing static demand is particularly valuable for long-duration deployments because the energy accumulates continuously. Reducing dynamic demand is important when active operations are frequent or computationally intensive. Power gating, peripheral shutdown, clock management, and deep-sleep states can address background consumption, while efficient algorithms and communication scheduling can reduce dynamic consumption. A complete energy model should therefore identify both categories and quantify their contributions.

The energy cost of a sensor event can vary significantly with the amount of information generated. A low-frequency temperature measurement may require only a small packet, whereas a microphone or accelerometer can produce a high-volume stream. Signal-rich sensors can therefore benefit strongly from local feature extraction, compression, event detection, or adaptive sampling. The design objective is to retain application-relevant information while discarding redundancy. This principle links energy efficiency with information processing: the node should spend energy on measurements and computations that contribute directly to the application's decision. In future IoT systems, energy-aware information theory and task-oriented sensing may provide useful foundations for determining which data should be acquired, processed, stored, and transmitted.

4. Power Profiling, Measurement, and Energy Analysis

Power profiling is a systematic method for understanding how an IoT node consumes energy over time. The supplied material recommends measuring and recording sensor activity, data transmission events, timestamps, data sizes, and power values. A basic profile should distinguish at least sleep, wake-up, sensing, processing, transmission, and reception states. Current can be sampled using a precision current monitor, shunt resistor with a suitable amplifier, or a dedicated power analyzer. The measurement system must have sufficient bandwidth to capture short radio bursts and wake-up transients; otherwise, important energy events may be missed. Voltage should also be recorded because energy is the integral of power over time. A current-only measurement is insufficient when the supply voltage changes substantially across operating states.

Energy profiling should be connected to application behavior. For example, a temperature sensor sampled every minute and transmitting every minute has a very different profile from the same sensor sampled locally every second but transmitting only when a threshold is crossed. The profile should therefore include the sensing interval, processing interval, packet size, transmission interval, retransmission rate, sleep duration, and wake-up latency. From these measurements, average current and daily energy can be estimated. If E_{event} represents the energy of one complete sensing-and-communication cycle and N is the number of cycles per day, daily energy can be approximated by $E_{\text{day}} = N \times E_{\text{event}} + E_{\text{background}}$. This approach allows researchers to compare alternative schedules and determine whether energy savings come from hardware, software, communication, or reduced activity.

Energy breakdown analysis is useful for identifying the dominant source of consumption. A node may spend only a small fraction of its time actively sensing but still consume most of its energy during radio operation. Alternatively, a gas sensor with a continuously powered heater may dominate the energy budget even when communication is infrequent. The source material recommends separating sensing, processing, communication, and sleep-related energy and examining power spikes associated with specific operations. Correlation analysis can reveal whether energy rises with sampling rate, packet size, processing complexity, or retransmission count. Such evidence is more valuable than a single average-current number because it identifies the design variable that should be optimized.

Sleep and wake-up analysis is particularly important for battery-powered systems. Duty cycling places the node in a low-power state for a defined period and activates it only when measurements or communications are required. The energy benefit depends on the sleep current, wake-up energy, wake-up time, and fraction of time spent asleep. If the wake-up process is expensive or too frequent, aggressive duty cycling may not provide the expected benefit. Event-triggered mechanisms can improve efficiency when events are sparse, whereas periodic monitoring may require predictable wake-up schedules. Adaptive duty cycling can combine both approaches by changing the sampling or wake-up interval according to context. For example, stable environmental conditions may permit long intervals, while rapid changes can temporarily increase sampling frequency.

A publishable experimental methodology should report the instrumentation, sampling rate, supply voltage, measurement uncertainty, number of repetitions, environmental conditions, firmware version, communication configuration, and workload. Researchers should distinguish between component-level measurements and complete-node measurements. They should also report energy per useful sample, energy per transmitted byte, energy per successful event, and estimated operational lifetime where appropriate. This level of detail supports reproducibility and prevents misleading comparisons. The source material provides a conceptual foundation for power profiling, while contemporary reviews emphasize the importance of application-specific trade-offs and network lifetime. [3]–[7] A standardized profiling methodology can therefore become an important contribution in experimental IoT research, especially when different sensors and communication technologies are compared under identical workloads.

Power profiling can be extended from descriptive measurement to predictive modelling. Once energy is measured for representative operating states, a workload model can estimate the energy required under different sampling intervals, packet sizes, and communication schedules without repeating every physical experiment. For example, a state-

transition model can estimate the number of wake-ups, active durations, and transmission events during a day. Such models can be validated against measured traces and then used for design-space exploration. This approach is useful when many sensor configurations must be compared. It also supports battery-lifetime prediction and helps determine whether an energy-saving strategy will remain beneficial when the workload changes.

Measurement uncertainty should be treated as part of the experimental methodology. Current measurement equipment introduces its own resolution, bandwidth, sampling limitations, and calibration uncertainty. Very short radio bursts may be difficult to capture accurately using a slow multimeter, while a high-bandwidth instrument may generate large datasets that require careful analysis. Researchers should therefore choose instrumentation according to the temporal scale of the energy events being studied. Repeated measurements under controlled conditions can provide estimates of variability. Reporting these details improves confidence in energy comparisons and prevents overinterpretation of small differences that may fall within the measurement uncertainty.

Power profiling can also support predictive maintenance by revealing changes in a node's normal energy signature. An increase in average current, wake-up duration, retransmission frequency, or sleep leakage may indicate sensor degradation, battery aging, communication problems, or hardware faults. Monitoring these indicators allows an IoT platform to detect abnormal energy behavior before complete failure occurs. Thus, energy telemetry can serve two purposes: optimizing consumption and assessing device health. Future experiments can investigate lightweight energy-health indicators that can be transmitted occasionally without creating significant additional overhead. Such indicators would be particularly valuable in large remote deployments where physical inspection is costly.

5. Energy-Efficient IoT Sensor Design Strategies

Energy-efficient design begins with requirements engineering. The first step is to define what must be measured, how often it must be measured, how quickly changes must be detected, how accurately the variable must be represented, and how long the node must operate without maintenance. Once these requirements are known, the designer can select a sensing principle and device that meets the application needs without excessive capability. The source material stresses the importance of balancing functionality, data accuracy, and power consumption. This balance is central to sustainable IoT design because reducing power by simply lowering measurement quality may produce an unusable system. A better strategy is to eliminate unnecessary energy expenditure while preserving the minimum performance needed by the application.

Low-power hardware selection is the next layer. Microcontrollers should provide suitable sleep modes, fast wake-up, efficient clocking, and peripherals that can operate without waking the main processor. Sensor devices should support low-power standby modes where possible, and communication modules should expose configurable transmit power and sleep states. Power regulators should be selected according to their quiescent current and expected load profile. Board-level components such as LEDs, voltage dividers, pull-up resistors, and USB interfaces should not be ignored because they can become significant in ultra-low-power designs. Miniaturization creates additional constraints because smaller nodes have limited energy storage and thermal capacity. The source material identifies miniaturization, hardware design, processing capability, and power constraints as persistent challenges.

Adaptive sensing can provide large energy savings when environmental conditions are predictable or events are sparse. Instead of sampling at a fixed high rate, a node can adjust the rate according to recent measurements, detected variability, or external context. A stable temperature environment may require only occasional measurements, whereas rapid changes may justify more frequent sampling. Threshold-based wake-up can also activate expensive sensors only when a low-power preliminary detector identifies a relevant event. Multi-stage sensing is therefore attractive: an inexpensive always-on detector provides coarse context, and a higher-energy sensor is activated only when necessary. Machine-learning methods can extend this concept by predicting when informative samples are likely to occur, although model complexity must be considered within the energy budget.

Data reduction is another important strategy. Transmitting every raw measurement can waste energy when successive readings contain little new information. Local filtering, aggregation, compression, feature extraction, and event reporting can reduce the number or size of wireless messages. Edge computing is particularly relevant because processing data near the sensor can reduce the distance and amount of communication to the cloud. The source material identifies data fusion, compression, sparse sensing, and edge computing as emerging approaches for energy reduction. The design challenge is to ensure that local processing does not consume more energy than the communication it replaces. A useful evaluation should compare the energy of local computation against the energy saved by reduced transmission under the same application workload.

Power management should operate as a closed loop. The node can monitor its energy state, workload, environmental context, and communication conditions and then adapt sampling, processing, transmission power, and sleep duration. Energy harvesting can complement this approach by supplying energy from solar, thermal, kinetic, or other ambient sources. Harvested energy is not automatically sufficient; the system must account for intermittency, storage efficiency, conversion losses, and load variability. Energy-aware algorithms can schedule high-energy tasks when harvested energy is abundant and postpone noncritical work during energy shortages. This combination of adaptive workload control and energy harvesting provides a path toward long-lived or self-sustaining sensor nodes.

Cross-layer optimization can provide greater benefits than isolated component-level optimization. A sensor's sampling policy affects the amount of data processed; processing affects the amount transmitted; communication affects radio active time; and radio activity affects how long the node must remain awake. These dependencies mean that a change in one layer can create savings or costs elsewhere. For example, increasing local compression may reduce radio energy but increase processor energy. Similarly, reducing the sampling rate may save energy but increase the risk of missing short-lived events. Cross-layer design evaluates these interactions together and selects configurations that satisfy the application's quality requirements with minimum total energy.

Power-aware firmware is equally important. Efficient firmware should initialize peripherals only when required, disable unused modules, avoid unnecessary memory transfers, use hardware peripherals when they are more efficient than software routines, and enter the deepest safe sleep state as soon as possible. Wake-up sources should be carefully selected so that the node can respond to important events without keeping the main processor active continuously. Firmware should also avoid excessive logging, debugging output, and repeated network discovery in production operation. These software-level decisions can have a substantial effect on battery life, particularly in systems where the sensing element itself is already highly optimized.

Design verification should be performed under realistic worst-case and typical-case workloads. A sensor that appears efficient during laboratory sleep conditions may consume considerably more energy when exposed to frequent events, poor wireless connectivity, or environmental extremes. Testing should therefore include representative idle periods, normal activity, peak activity, communication failures, and recovery conditions. The resulting measurements can be used to construct an energy budget with margins for uncertainty. This approach helps prevent under-sized batteries and ensures that the expected lifetime is achievable in actual deployment conditions.

6. Communication, Edge Processing, and Data Reduction

Wireless communication should be selected as an application-specific energy decision rather than a default networking choice. Short-range technologies such as BLE and Zigbee may be appropriate for local sensor networks, while LoRaWAN and other low-power wide-area approaches can serve applications requiring long-range communication with small payloads. Cellular IoT technologies can provide broad coverage but may impose different energy and connection-management requirements. Recent comparative work shows that energy performance depends on network technology, frequency band, data rate, payload size, distance, and protocol overhead rather than on protocol names alone. [4] Therefore, a fair comparison should use the same application traffic, reliability target, and measurement

methodology. The objective is to minimize energy per useful and successfully delivered piece of information, not merely radio current during a single transmission.

Communication scheduling is often as important as protocol selection. Batching several readings into one message can reduce repeated radio start-up and connection overhead, although excessive batching increases latency and memory requirements. Data aggregation can combine readings from multiple sensors or nodes before transmission. Adaptive transmit power can reduce energy when link conditions are favorable, while retransmission control can prevent repeated attempts under poor connectivity. The source material emphasizes reduced transmission frequency, data compression, and communication optimization as practical measures. At the network level, routing should consider residual energy, link quality, traffic load, and application deadlines. A route that minimizes hop count may not always minimize energy if it uses weak links that generate retransmissions.

Edge computing can further reduce communication energy by moving selected processing tasks closer to the sensor. Instead of sending raw sound, high-frequency motion streams, or dense environmental samples to a remote cloud, a local processor can extract events, features, or summaries. The resulting data may be much smaller. However, edge processing introduces computational energy and may require a more capable processor. The optimal partition therefore depends on workload intensity and radio characteristics. For simple measurements, transmitting raw values may be cheaper than complex local analysis; for high-volume signals, local feature extraction may be substantially more efficient. This trade-off should be evaluated experimentally using energy per application-level result.

Security must also be incorporated into energy-aware communication. Encryption, authentication, key management, secure boot, integrity checks, and retransmission protection consume computational and communication resources. A system that minimizes energy by disabling security mechanisms may be unsuitable for real deployments. Research on energy-efficient IoT security emphasizes the need to balance security services with constrained device resources. [3] Secure communication should therefore be designed with lightweight protocols, efficient session management, hardware acceleration where available, and appropriate communication scheduling. The energy cost of security should be measured as part of the complete workload so that designers understand the actual trade-off rather than treating security as an independent feature.

Communication and edge computing should ultimately be managed together. A sensor node can decide whether to transmit immediately, aggregate locally, perform additional computation, or wait for a favorable communication condition. Such decisions can be driven by application urgency, battery state, link quality, and harvested energy. This creates an adaptive architecture in which the communication subsystem is no longer operated at a fixed configuration. The source material's emphasis on energy-aware algorithms, adaptive sensing, data compression, and edge computing supports this integrated approach. Future IoT systems can extend it with learning-based policies that predict traffic demand, energy availability, and environmental changes while enforcing explicit reliability and latency constraints.

Network topology influences energy consumption in addition to protocol selection. Star, tree, mesh, and multi-hop architectures create different communication and forwarding requirements. In a mesh network, intermediate nodes may spend additional energy forwarding traffic, but the architecture can provide alternative routes and improve coverage. A star architecture can simplify management but may require longer direct transmissions. Cluster-based approaches can aggregate data and distribute forwarding responsibilities. The appropriate topology depends on node density, geographic area, reliability requirements, and traffic patterns. Energy-aware routing can incorporate residual battery level and link quality so that heavily loaded nodes are not repeatedly selected as forwarding points.

Communication efficiency should also be evaluated at the application level. A protocol that sends fewer packets is not necessarily better if those packets contain insufficient information or require costly retransmissions. Useful metrics include energy per successfully delivered application event, energy per valid sample, and energy per byte of meaningful information. This perspective encourages designers to consider semantic compression, event reporting, and application-aware aggregation. It also provides a common basis for comparing heterogeneous technologies. As

IoT networks increasingly combine short-range, wide-area, and edge communication, such application-level metrics will be more informative than isolated radio current specifications.

An additional consideration is firmware and protocol overhead associated with network discovery, synchronization, acknowledgements, connection establishment, and security handshakes. These operations may be infrequent, but their energy cost can become significant when devices repeatedly lose connectivity or restart. Persistent or scheduled connections, efficient session management, and local buffering can reduce repeated overhead when application requirements permit. At the same time, buffering should not introduce unacceptable data-loss risk. Communication optimization should therefore consider reliability, latency, memory usage, and recovery behavior together with energy.

7. Integration with Environmental Monitoring Systems

Environmental monitoring illustrates the value of integrating multiple IoT sensor types. A single sensor provides a narrow observation, whereas a network can combine temperature, humidity, pressure, light, gas, sound, and other measurements to describe a broader environmental state. The supplied material emphasizes data fusion with weather stations, satellite imagery, and IoT sensors; real-time monitoring and alerting; scalability; cross-domain correlation; improved resource management; advanced analytics; decision support; and open data. Such integration can improve the interpretation of sensor readings because environmental variables are often interdependent. For example, temperature and humidity can affect air-quality measurements, while weather conditions can influence pollutant concentration and sensor behavior. However, heterogeneous sensing also increases system complexity, communication traffic, calibration requirements, and energy demand.

Data fusion can improve energy efficiency when it is used to reduce redundant sensing and communication. Correlated measurements may be summarized locally rather than transmitted independently. A gateway can combine readings from nearby nodes and forward only significant changes or aggregated features. Context-aware fusion can also determine which sensors are necessary at a given time. If light levels indicate daylight, for example, an indoor system may reduce the sampling frequency of artificial-light-related measurements. Similarly, an environmental monitoring station can activate high-energy gas analysis only when low-energy indicators suggest an abnormal condition. These approaches transform the sensor network from a collection of independent devices into a coordinated sensing system.

Real-time alerting creates a different energy requirement from long-term environmental logging. Safety-critical events may require immediate communication and high sampling rates, whereas routine background monitoring can tolerate long intervals. A useful architecture therefore defines priority levels for sensor data. Critical events can bypass batching and compression delays, while noncritical measurements can be aggregated. The node can also adjust its energy policy based on the severity of detected conditions. This is an example of application-aware energy management: energy is spent where it provides the greatest information or operational value. Such policies should be explicitly specified because otherwise a generic sleep strategy may introduce unacceptable detection delays.

Environmental deployments also require careful consideration of sensor reliability and calibration. Long-term exposure to humidity, temperature variation, dust, chemicals, or vibration can alter sensor behavior. An energy-efficient node that produces inaccurate measurements is not a successful system. Calibration and health monitoring therefore need to be included in the energy budget. Self-diagnostics can be scheduled less frequently during stable conditions and more frequently when measurements become inconsistent. Redundant sensors can support fault detection, but they also consume energy. Cross-domain correlation provides an opportunity to identify faulty readings without continuously operating every redundant sensor at maximum sampling rate.

Integration with external environmental systems creates opportunities for decision support and sustainable resource management. Historical sensor data can be combined with predictive models to forecast environmental changes, optimize irrigation, manage building energy, detect pollution events, or support emergency response. Edge analytics can provide immediate local decisions, while cloud analytics can perform longer-term modelling. Open data can encourage collaboration among researchers and public agencies, but privacy, security, data quality, and

interoperability must be addressed. The source material emphasizes that integrated environmental monitoring can provide comprehensive coverage and informed decision-making. From an energy perspective, the objective is to create a hierarchy in which sensing, communication, and analytics are allocated to the layer where they can be performed with acceptable energy and latency.

Environmental monitoring systems must also address spatial and temporal heterogeneity. Sensors positioned only a few metres apart can observe different micro-environments, while conditions can change rapidly during weather events. A uniform sampling policy may therefore waste energy in stable locations and under-sample rapidly changing areas. Spatially adaptive deployment can activate or sample nodes according to detected variability. Temporal adaptation can increase measurement frequency during transitions and reduce it during stable periods. When combined with data fusion, these methods can preserve coverage while reducing redundant measurements. Such strategies are especially relevant to agriculture, air-quality monitoring, water monitoring, and smart-building applications where environmental conditions are not uniformly distributed.

Data governance becomes important when environmental sensor networks are integrated with public or organizational information systems. Sensor data may be shared among researchers, administrators, and automated decision systems, requiring consistent metadata, timestamps, calibration information, and quality indicators. Data provenance is valuable because decisions based on faulty or poorly calibrated sensors can be misleading. Secure access and integrity protection are also necessary when sensor data influence operational decisions. Energy-aware systems should therefore transmit not only measurements but, where appropriate, compact quality indicators and health status. This allows downstream systems to distinguish genuine environmental changes from sensor faults without requiring continuous high-power diagnostics.

Integration can also support coordinated maintenance. If several related sensors report inconsistent measurements, the system can identify a probable calibration or hardware issue and schedule maintenance for the affected location rather than inspecting every node. Energy-aware diagnostics can operate at a lower frequency during stable operation and increase temporarily when anomalies appear. This strategy combines environmental intelligence with device-health monitoring and can reduce unnecessary diagnostic traffic. It also supports more reliable long-term datasets, which are essential for scientific studies and policy decisions based on environmental observations.

8. Challenges and Emerging Research Directions

Despite rapid progress, energy-efficient IoT sensor design continues to face fundamental constraints. Battery capacity remains limited, and replacement can be expensive or impossible in remote deployments. Miniaturization reduces physical space for batteries, power-management components, and thermal dissipation. Resource-constrained processors may not support sophisticated analytics, while real-time applications cannot always tolerate long sleep periods. Wireless communication introduces unpredictable energy costs because channel conditions, interference, retransmissions, and network availability vary over time. The supplied source identifies power constraints, miniaturization, processing limitations, communication efficiency, environmental variability, real-time operation, legacy integration, cost, and standardization as major challenges. These issues demonstrate that energy efficiency cannot be achieved through one technique; it requires coordinated hardware, firmware, communication, and application design.

Environmental energy harvesting is a promising direction, but it is not a universal replacement for batteries. Solar harvesting depends on illumination, thermal harvesting depends on temperature gradients, and kinetic harvesting depends on mechanical activity. Harvested energy can be intermittent and may not match the timing of application demand. Effective systems therefore require energy-aware scheduling and suitable storage. A sensor can use a small energy buffer to accumulate harvested energy and then execute a high-energy operation when sufficient energy is available. This approach changes the design objective from minimizing instantaneous power to maintaining energy neutrality over time. Research in sustainable IoT increasingly considers energy harvesting, storage, routing, and

communication together. [6] The challenge is to make these systems robust under variable and uncertain energy availability.

Artificial intelligence and machine learning offer another emerging direction. AI can predict energy availability, classify sensor events, estimate environmental dynamics, select sampling rates, and optimize communication schedules. However, AI models themselves consume energy and memory. A large model running continuously on a small microcontroller may negate the intended energy savings. TinyML and model compression therefore become relevant, with techniques such as quantization, pruning, feature selection, and lightweight inference reducing computational cost. AI should be introduced only when its predictive benefit exceeds its energy overhead. A useful research question is not simply whether an AI method improves accuracy, but whether it improves application-level utility per joule compared with a simpler rule-based strategy.

Security, interoperability, and sustainability are increasingly connected. Secure operation requires computation and communication, while heterogeneous platforms can create duplicated functionality and inefficient data exchange. Standardized interfaces and lightweight protocols can reduce integration overhead and improve maintainability. Security mechanisms should be energy-aware but should not be removed merely to extend battery life. Recent research also highlights the joint consideration of energy and security trade-offs in sustainable IoT systems. [8] Future designs are therefore likely to optimize several objectives simultaneously: energy consumption, security strength, latency, accuracy, reliability, and carbon impact. Multi-objective optimization can provide a more realistic framework than single-metric energy minimization.

Another future direction is self-adaptive sensor architecture. Instead of configuring a node once during deployment, the system can continuously observe its own energy state, environmental variability, communication quality, and application demand. It can then modify sampling intervals, sensor activation, processor frequency, radio parameters, and data-reduction methods. Such adaptation must be bounded by safety and quality-of-service constraints. Explainability and predictability are also important when automated policies control critical infrastructure. The long-term research opportunity is therefore a closed-loop energy management system that combines accurate power profiling, context-aware sensing, edge intelligence, energy harvesting, secure communication, and application-level quality metrics. This direction is consistent with the source material's emphasis on energy-aware algorithms, context-aware sensing, edge computing, and emerging low-power technologies.

Interoperability is another major research issue because IoT deployments frequently combine components from different vendors and generations. A legacy sensor may use a proprietary interface while a newer node uses a low-power digital protocol. Gateways can bridge these technologies, but gateways themselves consume energy and may become single points of failure. Standardized data models, lightweight messaging, and modular interfaces can reduce integration complexity. Future research should therefore evaluate energy efficiency together with interoperability overhead. A technically efficient sensor that requires extensive protocol conversion or frequent gateway interaction may not be optimal at system scale.

Sustainability also requires considering the full lifecycle of sensor hardware. Manufacturing, transportation, battery production, maintenance, replacement, and disposal contribute to environmental impact. Long battery life reduces replacement frequency, but repairability, modularity, recyclable materials, and responsible end-of-life handling can further improve sustainability. Energy harvesting may reduce battery dependence, but harvesting components can add material and manufacturing cost. A lifecycle perspective therefore complements operational energy measurements. Future studies could report energy consumption alongside maintenance events and hardware replacement rates to provide a more complete assessment of sustainable IoT deployment.

Standardized benchmarking is another open need. Different studies often report energy consumption using different supply voltages, workloads, communication distances, measurement instruments, and definitions of active or sleep current. This makes direct comparison difficult. A community-agreed benchmark could define representative sensing

workloads, communication patterns, environmental conditions, and reporting metrics. Such a benchmark would enable researchers to compare sensors and energy-management techniques more consistently and would help practitioners select technologies using evidence that is relevant to deployment conditions.

9. Proposed Evaluation Framework and Research Discussion

To make energy-efficiency claims comparable, this paper proposes an evaluation framework that combines electrical, sensing, communication, and application metrics. The first dimension is energy: average power, peak power, energy per sensing cycle, energy per successfully transmitted message, daily energy consumption, and estimated battery lifetime. The second dimension is sensing quality: accuracy, resolution, sampling completeness, false-event rate, and detection latency. The third dimension is communication: packet delivery ratio, retransmission rate, latency, payload size, and energy per useful byte. The fourth dimension is system reliability: uptime, fault rate, recovery time, and resilience to environmental variation. Finally, sustainability can include battery replacement frequency, harvested-energy contribution, and estimated operational carbon impact. Reporting multiple dimensions prevents an energy-saving method from appearing successful when it significantly degrades application performance.

An experimental comparison should define a common workload for all sensor candidates. For example, temperature sensors can be evaluated under identical sampling intervals, supply voltage, communication schedules, and environmental conditions. The experiment should include both steady-state and event-driven periods so that the effect of adaptive strategies can be observed. Each configuration should be repeated sufficiently to account for measurement variation. Results should be reported with appropriate descriptive statistics and, where relevant, confidence intervals. If a communication protocol is compared with another protocol, payload size, reporting frequency, link distance, and reliability requirements should be controlled. Such discipline is essential because energy measurements are highly sensitive to workload and configuration.

Battery-lifetime estimation can be derived from measured energy rather than nominal sensor current. If a battery provides an effective energy capacity E_{batt} and the measured average daily energy requirement is E_{day} , a first-order lifetime estimate is $L_{\text{days}} = E_{\text{batt}} / E_{\text{day}}$. Real deployments require derating because of battery discharge characteristics, temperature, self-discharge, regulator losses, aging, and peak-current limitations. For energy-harvesting systems, the evaluation should additionally report harvested energy per day, storage efficiency, and the fraction of application demand satisfied by harvested power. These metrics allow researchers to distinguish between nominally self-powered operation and genuinely energy-neutral operation.

Table 3 provides a proposed reporting template for future experiments. The framework is intentionally application-neutral so that it can be used for environmental monitoring, smart buildings, industrial sensing, wearables, agriculture, and mobility applications. Researchers can extend the framework with domain-specific metrics. For example, a healthcare sensor may require clinical detection sensitivity, whereas an industrial sensor may require deterministic latency and fault tolerance. The key principle is that energy should be evaluated in relation to useful system output. A sensor that uses slightly more energy but produces significantly more reliable information may be preferable to a lower-power alternative. Conversely, a high-performance sensor that provides unnecessary precision may waste energy without improving application value.

The proposed framework also supports lifecycle-oriented design. Energy consumption should be measured not only during active operation but across deployment, maintenance, battery replacement, firmware updates, and end-of-life considerations. Large-scale IoT systems can contain many small energy consumers, so maintenance logistics can become an important part of total operational cost. Energy-efficient design can reduce the frequency of physical interventions and thereby improve scalability. The framework can therefore support future comparative studies, design-space exploration, and multi-objective optimization. It converts the broad concept of energy efficiency into a set of measurable indicators that can be reported consistently in journal research.

Dimension	Recommended metrics	Interpretation
Energy	Average power; peak power; energy/cycle; daily energy	Lower is generally better within QoS limits
Lifetime	Estimated battery life; replacement interval	Longer is generally better
Sensing	Accuracy; resolution; sampling completeness	Meet application requirement
Communication	Delivery ratio; retransmissions; energy/useful byte	High reliability with low energy
Latency	Wake-up + processing + transmission delay	Meet application deadline
Reliability	Uptime; fault rate; recovery time	Higher reliability
Sustainability	Harvested-energy contribution; maintenance burden	Lower external energy and maintenance
Adaptability	Response to workload/environment changes	Stable QoS under changing conditions

Table 3. Recommended evaluation metrics for energy-efficient IoT sensor studies.

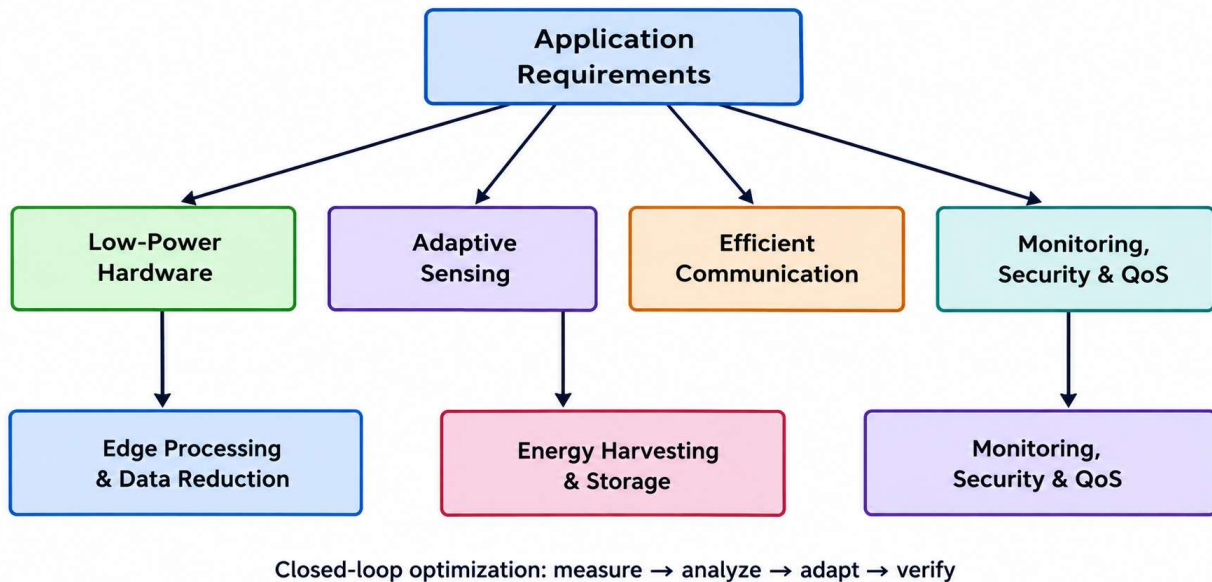


Figure 2. Proposed closed-loop framework for energy-efficient IoT sensor design and evaluation.

A further benefit of the evaluation framework is that it supports multi-objective optimization. Energy, accuracy, latency, reliability, and cost rarely improve simultaneously. A configuration that minimizes energy may reduce responsiveness, while a configuration that maximizes accuracy may require continuous sensing and high communication overhead. Pareto analysis can identify configurations for which no objective can be improved without

degrading another. Researchers can then select a design according to the priorities of the application. This is more realistic than reporting a single energy number because real IoT systems operate under competing constraints.

Future experiments should also include baseline configurations. A baseline can be a continuously active sensor, a fixed-rate sampling policy, or a conventional communication schedule. Energy-saving strategies can then be compared against this reference using percentage reduction, lifetime extension, and changes in sensing quality. Statistical testing may be appropriate when multiple configurations produce close results. Reporting both absolute and relative values is recommended because a large percentage saving from a very small baseline may have limited practical significance. Such reporting will make journal studies easier to reproduce and compare.

An energy-efficiency study should also document the baseline hardware and software configuration. Processor type, clock frequency, memory configuration, sensor interface, radio settings, firmware version, sampling interval, packet format, and power supply should be stated clearly. If a gateway or cloud service is used, its contribution should be separated from the sensor-node measurement unless the study explicitly evaluates end-to-end energy. This separation allows readers to determine whether improvements originate from the sensor node, the communication layer, or the broader infrastructure. Clear baselines and transparent reporting are therefore essential for credible journal publication.

10. Conclusion

IoT sensors serve as fundamental components that bridge the physical and digital worlds by continuously capturing information such as temperature, humidity, pressure, motion, proximity, light, acceleration, location, gas, and sound. Their widespread use in smart homes, agriculture, healthcare, industrial monitoring, and environmental management demonstrates their importance in modern intelligent systems. However, effective sensor selection cannot be based solely on measurement capability. Power consumption, operating modes, processing requirements, communication overhead, environmental conditions, reliability, maintenance, and application requirements must be considered together to achieve sustainable and dependable IoT deployments.

Energy efficiency is therefore a critical architectural objective in IoT sensor design. Energy is consumed during sensing, processing, communication, idle operation, sleep transitions, and supporting circuitry. Techniques such as duty cycling, adaptive sampling, event-driven sensing, low-power hardware, edge processing, data reduction, efficient communication, and energy harvesting can substantially improve operational lifetime. Nevertheless, optimization should maintain an appropriate balance between energy consumption, sensing accuracy, latency, reliability, and quality of service rather than simply minimizing power consumption.

Environmental monitoring further demonstrates the importance of heterogeneous sensor integration, data fusion, and adaptive operation. Future IoT systems should dynamically respond to changes in environmental conditions, workload, available energy, and communication quality. Accurate power profiling and complete-node experimentation are essential because datasheet values may differ according to operating modes, supply conditions, hardware implementation, and workload.

Overall, sustainable IoT sensing requires coordinated optimization across hardware, firmware, communication, edge intelligence, security, and energy management. The proposed evaluation framework provides a systematic basis for assessing energy consumption, lifetime, sensing quality, reliability, latency, sustainability, and adaptability. Future research should focus on closed-loop, application-driven architectures that continuously **measure, analyze, adapt, and verify** system behavior. Such approaches can reduce maintenance requirements, improve scalability, extend device lifetime, and enable more environmentally responsible IoT deployments.

References

- [1] D. Miorandi, S. Sicari, F. De Pellegrini, and I. Chlamtac, “Internet of things: Vision, applications and research challenges,” *Ad Hoc Networks*, vol. 10, no. 7, pp. 1497–1516, 2012. doi: 10.1016/j.adhoc.2012.02.016.
- [2] T. Rault, A. Bouabdallah, and Y. Challal, “Energy efficiency in wireless sensor networks: A top-down survey,” *Computer Networks*, vol. 67, pp. 104–122, 2014. doi: 10.1016/j.comnet.2014.03.027.
- [3] H. Hellaoui, M. Koudil, and A. Bouabdallah, “Energy-efficient mechanisms in security of the internet of things: A survey,” *Computer Networks*, vol. 127, pp. 173–189, 2017. doi: 10.1016/j.comnet.2017.08.006.
- [4] L. Mekki et al., “Energy Efficiency in Short and Wide-Area IoT Technologies—A Survey,” *Future Internet*, vol. 13, no. 1, 2021. doi: 10.3390/fi13010022.
- [5] S. Benhamaid, A. Bouabdallah, and H. Lakhlef, “Recent advances in energy management for Green-IoT: An up-to-date and comprehensive survey,” *Journal of Network and Computer Applications*, vol. 198, 103257, 2022. doi: 10.1016/j.jnca.2021.103257.
- [6] “A comprehensive review of energy harvesting and routing strategies for IoT sensors sustainability and communication technology,” *Sensors International*, vol. 5, 100258, 2024. doi: 10.1016/j.sintl.2023.100258.
- [7] “An analytical survey of energy efficiency in IoT paradigm,” *Procedia Computer Science*, vol. 210, pp. 283–288, 2022. doi: 10.1016/j.procs.2022.10.151.
- [8] K. A. Darabkh and W. K. Kassab, “Energy-Security Tradeoffs in Sustainable IoT Systems: A Systematic Survey of AI and Blockchain Solutions,” *Array*, 101205, 2026. doi: 10.1016/j.array.2026.101205.
- [9] “A survey and taxonomy on energy management schemes in wireless sensor networks,” *Journal of Systems Architecture*, vol. 111, 101782, 2020. doi: 10.1016/j.sysarc.2020.101782.
- [10] “A comprehensive review on energy efficient Internet of Things based wireless sensor network,” 2023 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS), IEEE, 2023. doi: 10.1109/ICCCIS60361.2023.10425545.