Optimizing Energy Efficiency in Battery-powered IoT Devices through Hardware Optimization and Voltage Scaling

A. R. Bhavya
School of Electronics and Communication Engineering, REVA University, Bangalore, India
r20pec06@ece.reva.edu.in (corresponding author)

K. M. Sudharshan
School of Electronics and Communication Engineering, REVA University, Bangalore, India
sudharshankm@reva.edu.in

Received: 18 November 2024 | Revised: 23 December 2024 | Accepted: 11 January 2025

 DOI: https://doi.org/10.48084/etasr.9640

ABSTRACT

With the rapid proliferation of battery-powered Internet of Things (IoT) devices, optimizing energy efficiency has become a critical challenge, especially in wireless communication modules. This work focuses on the power consumption analysis of the ESP32-WROOM module, a widely used wireless communication component in IoT applications. By evaluating five key interfaces, UART, I2C, SPI, I2S, and Wi-Fi, over different CPU clock frequencies (40 MHz, 80 MHz, 160 MHz, 240 MHz) and operating voltages (3 V and 3.3 V), this study provides a comprehensive understanding of how these parameters affect the energy efficiency in battery-powered IoT systems. The primary contribution of this work is the identification of critical trade-offs between clock frequency, voltage scaling, and power consumption across different interfaces. The findings reveal that higher CPU frequencies lead to increased power consumption across all interfaces, with I2S consuming the highest current (up to 64.6 mA) at 240 MHz and 3.3 V. Wi-Fi, often considered a power-intensive interface, showed significant current surges, particularly during connection establishment, with a peak current of 280 mA at 240 MHz and 3 V. These variations highlight the importance of effective voltage regulation during link establishment to mitigate power inefficiencies. Additionally, the voltage differential between 3 V and 3.3 V was found to influence overall power consumption, although certain interfaces at higher frequencies demonstrated marginal efficiency improvements when operating at 3.3 V. This highlights that while voltage selection is important, clock frequency adjustments have a more profound effect on power consumption. This work provides actionable insights for developers aiming to optimize power consumption in IoT applications. The findings provide guidance for selecting appropriate operating frequencies and voltage levels, contributing to significant energy savings and extended battery life in energy-constrained IoT and embedded systems.

Keywords-power consumption; Internet of Things; wireless module; frequencies; voltage levels; LDO

I. INTRODUCTION

One of the most critical issues to be addressed in the design of battery-powered Internet of Things (IoT) devices is power consumption, as such devices often have limited access to energy sources. Many IoT devices are located in hard-to-reach areas, such as agricultural regions, infrastructure structures, or as part of environmental monitoring techniques, where battery replacement or recharging is inconvenient and expensive [1]. For these reasons, there is a need to maximize the efficiency of energy utilization to extend the time that the battery can sustain the use of the devices. Many IoT devices need to perform several tasks such as data acquisition, data processing, and data transmission, and all these activities require energy. The goal is to be able to continue these operations for several years without having to recharge or replace the battery. In other words, one of the main concerns with high power consumption is that it causes the batteries to discharge quickly, which reduces the reliability of the system as well as the frequency of maintenance, which in turn increases the cost of ownership [2]. There are cases where battery life is important in the IoT design, such as in wearables, where long battery life is desirable and recharging the battery at frequent intervals may not be possible [3]. In smart city technology and industrial applications of IoT where thousands of connected devices are deployed and work as a network, power utilization is the main issue for the scalability and feasibility of the system [4]. Improving the performance of communication methods and sensors will increase the lifetime of IoT devices, minimize their downtime, and increase the application of IoT technology in various fields. The ESP32-WROOM has quickly become popular in IoT devices, especially battery-powered devices, due to its low power consumption, on-chip communication interfaces, and relatively low cost. The ESP32-WROOM is a highly integrated microcontroller with both Wi-Fi and Bluetooth capabilities, which are very useful in various IoT applications that require wireless connectivity, including smart home devices, wearables, and remote-control systems [5]. Their key feature is the utilization of low-power modes, namely deep sleep, hibernation, and active mode, where devices can minimize power consumption when not in use. For example, in deep sleep mode, the ESP32-WROOM can consume as little as a few microamps, greatly increasing battery life in cases where the device is turned on and off or spends most of its time in standby [6]. Also, the ESP32-WROOM incorporates dual-core processors that allow tasks to be efficiently divided between two cores, so that tasks that require the system to communicate or operate at high performance can be performed while the second core performs less significant tasks in the background to conserve power [7]. Another important feature is the ability to connect various sensors and peripherals, which opens up many possibilities for various applications and keeps the power consumption indicator to a minimum [8]. Well-funded libraries, development boards and other support, extensive communities of ESP32-WROOM have addressed the dynamic and energy-efficient IoT solutions sought by developers [9]. Therefore, power optimization in this microcontroller is one of the main tasks for further research, as it is perfectly suited for consumer and industrial IoT applications with a good balance of performance, connectivity, and power reduction. Some of the key contributions of this work are:

- Conducting a comprehensive power consumption analysis across multiple interfaces of the wireless device.
- Providing insights into power consumption at 3 V and 3.3 V, allowing flexibility in voltage selection based on application requirements.
- Optimizing power efficiency by evaluating the effects of varying CPU frequencies and operating voltages.
- Offering practical recommendations for developers to extend battery life through optimal selection of operating frequencies and voltages.
- Contributing to energy-saving strategies by identifying the optimal trade-off between power consumption and performance in embedded systems.

A literature review on power minimization in IoT devices reveals some gaps and highlights the need for further research. The adaptable buck converter allows its output voltage to be configured at runtime by the microcontroller. This feature enables easy integration of the module into circuits with varying voltage requirements, reduces energy consumption through voltage scaling, reduces component count, and consequently reduces system cost. [10]. Power is supplied and controlled by the controller and on-board, and the design must include mechanisms to turn off peripherals to avoid wasting idle energy. This is certainly the case when entering sleep mode. Low-current peripherals can be powered directly from IO pins. The microcontroller can then completely turn off the peripheral during sleep, eliminating all sensor sleep currents. For high-power hardware, such as radio modules, the IO ports will not be able to support the current drawn in active mode. Therefore, external digital load switches can be incorporated into the IoT node design. The TPS22860 is an example of an ultra-low leakage load switch that can be used in an IoT context. The leakage current of these switches is important to minimize the energy drain in sleep mode. Another approach, when using multiple voltage rails for multiple peripherals is to disable any unused voltage rail. By using an LDO with an enable pin, one could instantly turn off any peripherals connected to that voltage rail. Clearly, any microcontroller-based power management must be implemented through careful hardware and firmware design [11].

II. PROPOSED SYSTEM DESIGN

The purpose of this research is to analyze and improve the power consumption of battery-powered IoT devices, with a focus on the popular ESP32-WROOM microcontroller, by studying how this device operates at critical voltage levels. Voltage levels are significant for battery-powered IoT devices and are therefore critical for assessing energy consumption. In addition, the study will quantify the current present in each of the ESP interfaces, such as Wi-Fi, Bluetooth, GPIO, and sensors, at a given time under various operating conditions. The measurements will help understand the impact of different voltage levels on the overall power consumption and battery life. This knowledge will help developers determine the most efficient power management techniques so that the devices they create are energy efficient, yet powerful. The ultimate objective is to provide a comprehensive set of insights that developers can utilize to optimize the power consumption of IoT devices based on voltage and frequency tuning, thereby increasing battery life. The proposed block diagram of the ESP32–WROOM is depicted in Figure 1. In the proposed setup, two voltage converters, powered by a 4.7V lithium-ion battery, are customized according to the requirements. A 3.3 V output is generated by an LDO, and a 3 V output is generated by a low power buck converter. However, the 3 V can also be obtained by adding a low power LDO so that the system architecture has two separate 3 V and 3.3 V LDOs with common ground to power the controller and positive to VDD. The power can be dynamically varied on demand during system operation using the EN1/EN2 enable pins. The voltage can be varied between 3 V and 3.3 V without affecting the operation of the ESP32. Most semiconductor manufacturers' power management strategies focus on the chip level, leaving external hardware and firmware optimization relatively unexplored. In this study, the current consumption of the ESP32-WROOM module is analyzed at 3 V and 3.3 V at various frequencies, examining the effects on peripherals such as UART, I2C, SPI, I2S, and Wi-Fi. The results are then compared to highlight the power savings across these configurations. Note that the quiescent current of the buck converter should be kept as low as possible, in the range of pA to nA. Figure 2 shows the buck converter utilized to provide a regulated 3 V output from a battery voltage. The most crucial and central part of the circuit is the LM2596S-ADJ IC, which switches the inductor L1 to provide the required voltage conversion.

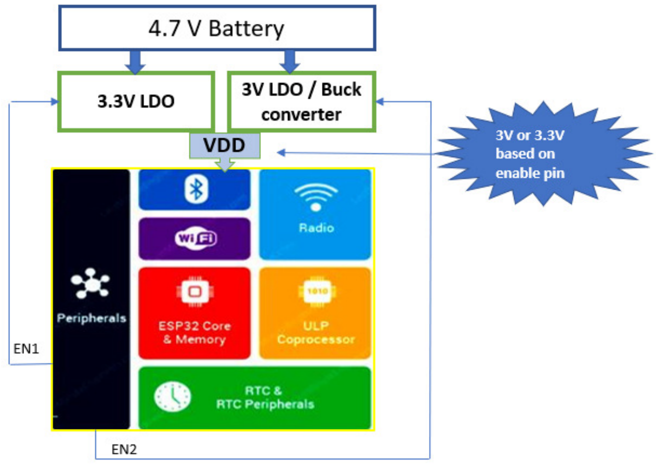


Fig. 1. Block diagram of the ESP32-WROOM module.

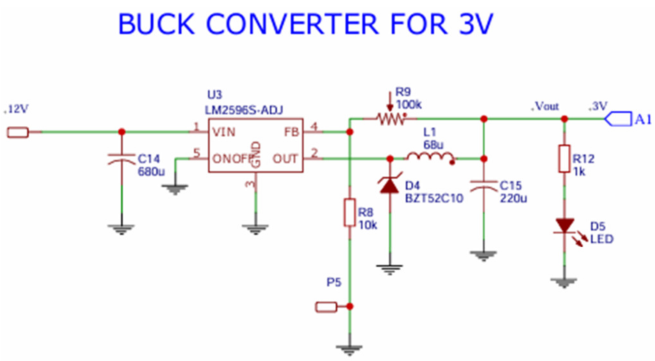


Fig. 2. Buck converter for 3 V voltage.

III. METHODOLOGY

Authors in [12] address the issue of the energy efficient operation of an ESP32 and propose a best practice for using an ESP32 in an industrial wireless sensor network. The different modes of operation of the ESP32 are mentioned and it is recommended to switch between operation modes over time to perform tasks with the appropriate mode. It is also noted that in active mode, energy efficiency can be further improved by adjusting the processor clock speed. However, there has been limited work addressing the requirements of microcontrollers for wearable IoT applications and optimizing communication to a local storage medium. This is certainly a niche area, but the steady growth, ease of access, and resulting variety of use cases have shown that the evaluation of further optimization methods is still useful and relevant. Therefore, our experiment evaluated the power consumption of the wireless device for different communication protocols, including Wi-Fi, UART, I2C, I2S, and SPI, at two voltage levels: 3 V and 3.3 V. The results support the use of lower voltage configurations to optimize power efficiency without significantly impacting communication performance.

A. UART

Table I and Figure 3 show the power consumption of the ESP32-WROOM UART interface at different CPU frequencies (40 MHz, 80 MHz, 160 MHz, and 240 MHz) and two supply voltages, 3 V and 3.3 V. The current consumption of the UART at 40 MHz frequency with 3 V and 3.3 V is shown in Figure 4. The voltage reduction from 3.3 V to 3 V has a significant impact on the power consumption and therefore increases the efficiency.

TABLE I. POWER CONSUMPTION OF ESP32-WROOM UART AT DIFFERENT FREQUENCIES AND VOLTAGES

Frequency	Power at 3 V (mW)	Power at 3.3 V (mW)	Power savings (%)
40 MHZ	41.7	47.19	11.63
80 MHZ	70.2	77.22	9.09
160 MHZ	95.4	105.93	9.9
240 MHZ	114	126.72	10.03

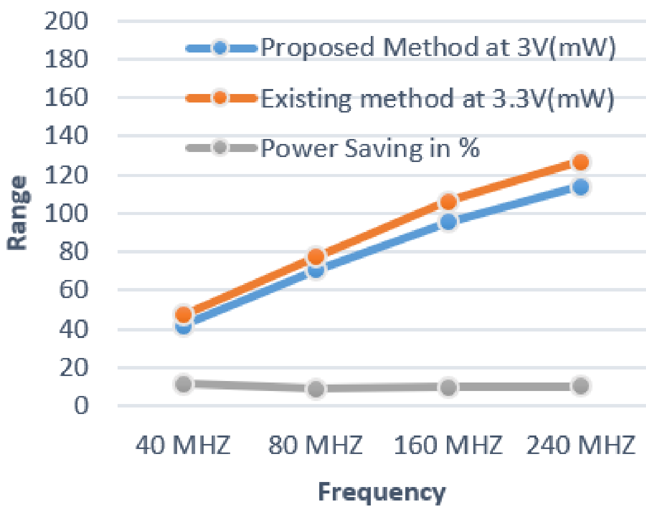
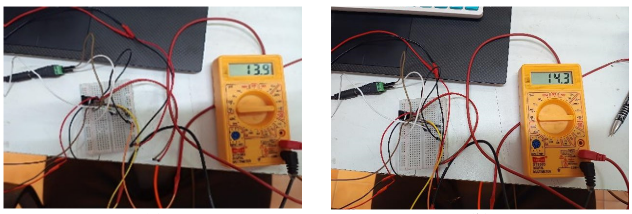


Fig. 3. Graph interpretation of UART power consumption.



(a) (b)

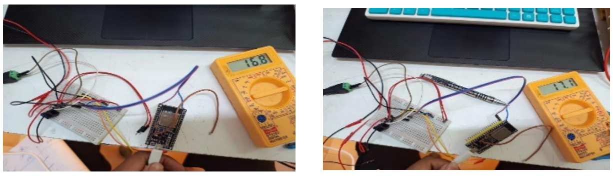
Fig. 4. Current consumption (mA) of the ESP32-WROOM UART at 40 MHz with: (a) 3 V and (b) 3.3 V.

B. I2C

Table II shows the power consumption of the I2C interface of the ESP32-WROOM at different CPU frequencies and voltages, where the power consumption is lower at the lowest frequency and at 3 V. The current consumption of the I2C at 40 MHz frequency with 3 V and 3.3 V is depicted in Figure 5.

TABLE II. POWER CONSUMPTION OF ESP32-WROOM I2C AT DIFFERENT FREQUENCIES AND VOLTAGES

Frequency	Power at 3 V (mW)	Power at 3.3 V (mW)	Power savings (%)
40 MHZ	50.4	56.43	10.6
80 MHZ	96.9	107.91	10.2
160 MHZ	135.3	149.82	9.69
240 MHZ	175.5	192.06	8.6



(a) (b)

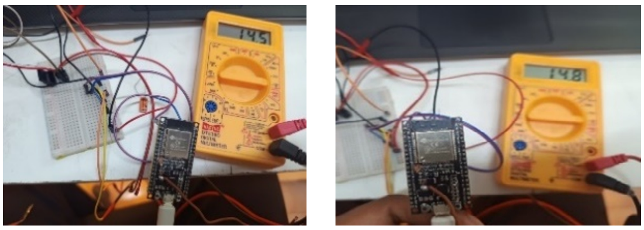
Fig. 5. Current consumption (mA) of the ESP32-WROOM I2C at 40 MHz with: (a) 3 V and (b) 3.3 V.

C. SPI

Table III shows the power consumption of the ESP32-WROOM SPI interface at different CPU frequencies (40 MHz, 80 MHz, 160 MHz, and 240 MHz) and two supply voltages, 3 V and 3.3 V. The current consumption of the SPI at 40 MHz frequency with 3 V and 3.3 V is shown in Figure 6. As expected, the power consumption increases as the frequency increases. However, as the current consumption remains almost the same, the power consumption decreases at 3V.

TABLE III. POWER CONSUMPTION OF ESP32-WROOM SPI AT DIFFERENT FREQUENCIES AND VOLTAGES

Frequency	Power at 3 V (mW)	Power at 3.3 V (mW)	Power savings (%)
40 MHZ	43.5	48.84	10.9
80 MHZ	78	87.45	10.80
160 MHZ	78.21	92.1	15.09
240 MHZ	115.5	128.04	9.7



(a) (b)

Fig. 6. Current consumption (mA) of the ESP32-WROOM SPI at 40 MHz with: (a) 3 V and (b) 3.3 V.

D. I2S

Similarly, Table IV shows the power consumption of the ESP32-WROOM I2S interface at different CPU frequencies and two supply voltages. The current consumption of the I2S at 40 MHz frequency with 3 V and 3.3 V is shown in Figure 7. The results again show that the power consumption increases gradually as the frequency increases. Overall, it can be seen that the higher the frequency, the more power is consumed as a function of frequency, with the difference between 3V and 3.3V being relatively small at all frequencies tested.

E. Wi-Fi

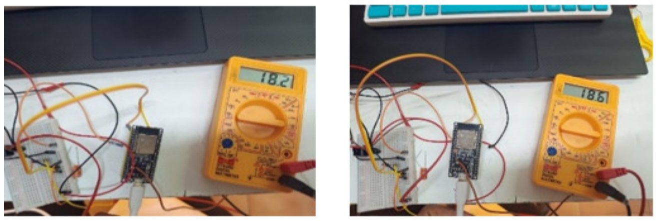
Table V shows the current consumption of the ESP32-WROOM Wi-Fi interface at different CPU frequencies and two supply voltages. The current consumption of the Wi-Fi at 160 MHz frequency with 3 V and 3.3 V is shown in Figure 8. At 160 MHz, the current consumption was 180 mA at 3 V during the connection attempts without achieving a successful connection. At 3.3 V the current consumption drops to 40 mA after connection. At 240 MHz, the device consumed 280 mA at 3 V during unsuccessful connection attempts, whereas at 3.3 V the device consumed 140 mA during connection and 50 mA after connection. In general, higher frequencies and voltages result in more stable Wi-Fi connections with high power surges during the connection phase. It can be seen that the Wi-Fi interface needs 3.3 V to achieve a stable connection as it won't connect at any frequency when the supply voltage is 3 V.

TABLE IV. POWER CONSUMPTION OF ESP32-WROOM I2S AT DIFFERENT FREQUENCIES AND VOLTAGES

Frequency	Power at 3 V (mW)	Power at 3.3 V (mW)	Power savings (%)
40 MHZ	54.6	61.38	11.04
80 MHZ	104.7	116.82	10.37
160 MHZ	148.5	165.66	10.3
240 MHZ	192	213.18	9.9

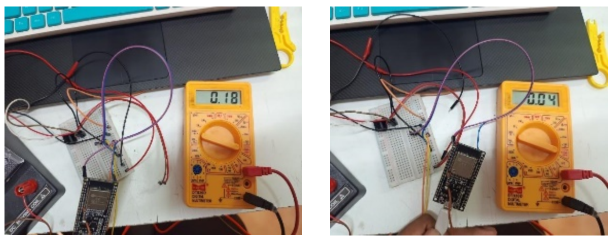
TABLE V. CURRENT CONSUMPTION OF ESP32-WROOM WI-FI AT DIFFERENT FREQUENCIES AND VOLTAGES

Frequency	Current at 3 V (mA)	Current at 3.3 V (mA)
40 MHZ	not connecting to Wi-Fi	
80 MHZ	- goes up to 270 mA - not connecting	- 110 mA while connecting - 38 mA after connecting
160 MHZ	- goes up to 180 mA - not connecting	- 120 mA while connecting - 40 mA after connecting
240 MHZ	- goes up to 280 mA - not connecting	- 140 mA while connecting - 50 mA after connecting



(a) (b)

Fig. 7. Current consumption (mA) of the ESP32-WROOM I2S at 40 MHz with: (a) 3 V and (b) 3.3 V.



(a) (b)

Fig. 8. Current consumption (A) of the ESP32-WROOM Wi-Fi at 160 MHz with: (a) 3 V and (b) 3.3 V.

IV. RESULTS AND DISCUSSION

The effect of the power consumption of the ESP32-WROOM module was considered at 3 V, 3.3 V and compared while interfacing with UART, I2C, SPI, I2S, and Wi-Fi peripherals at different CPU clock rates such as 40 MHz, 80 MHz, 160 MHz, and 240 MHz. The results indicate a clear trend: at 3.3V and higher frequencies, power consumption increases. At the same time, there is a significant reduction in power consumption of 8-15% at 3 V across all interfaces. The current consumption is greater at higher frequencies for all peripherals without being significantly affected by the voltage changes between 3 V and 3.3 V. The Wi-Fi power consumption fluctuated significantly, reaching its maximum during the connection attempts and was unstable during the connection at 3 V, and operated well at 3.3 V. These findings could be helpful in determining the factors that would lead to better efficiency of ESP32-WROOM used in power critical applications.

V. CONCLUSION

This study focuses on battery-operated Internet of Things (IoT) devices where energy availability is limited. Our experiment presents a power analysis of the ESP32-WROOM module, which is representative of most wireless devices. It focuses on interfaces such as UART, I2C, SPI, I2S, and Wi-Fi, with operating frequencies of 40 MHz, 80 MHz, 160 MHz, and 240 MHz, and voltages of 3 V and 3.3 V. The results show that the current remains almost the same for both 3 V and 3.3 V across all peripheral operations, which means that according to the formula $P = VI$, the power consumption is lower at 3 V. In addition, the power consumption for all interfaces increases with increasing CPU clock frequency, which is expected since higher frequencies lead to higher power consumption. The highest power consumption was observed for the I2S interface, which reached 64.6 mA at 240 MHz and 3.3 V. Wi-Fi also showed significant current consumption, especially during connection phases at 240 MHz. These findings emphasize the importance of developers carefully selecting appropriate clock frequencies and dynamically adjusting the voltage. The system should operate primarily at 3 V and lower frequencies, switching to 3.3 V and higher frequencies only when necessary, such as for Wi-Fi. Similarly, switching to higher frequencies and lowering the voltage to 3 V during high performance requirements will conserve energy and extend battery life. This approach could extend battery life by at least 15%, assuming the system operates at 3 V and lower frequencies. For instance, if the battery is expected to last 24 months, this strategy could add an additional 3-4 months for a total battery life of approximately 28 months. The key contributions of the proposed methodology include:

1. A quantitative analysis of power consumption across different interfaces, revealing that voltage scaling can result in power savings of up to 15%.
2. A framework for frequency and voltage scaling that provides actionable insights for optimizing power usage in battery-powered IoT devices.
3. Specific insights into the dynamic power fluctuations of Wi-Fi, providing strategies for energy management during communication phases.
4. Recommendations for developers to extend battery life by selecting the optimal frequency and voltage settings for specific application needs.

In conclusion, the proposed method offers valuable insights for optimizing energy efficiency in battery-powered IoT devices through strategic frequency and voltage adjustments. This approach can significantly extend battery life, reduce maintenance costs, and improve the sustainability of IoT systems in practical applications.

ACKNOWLEDGEMENT

The authors warmly acknowledged the Bangalore Institute of Technology, VTU,Bengaluru, Karnataka, India for providing the facilities required to carry out the research.

REFERENCES

[1] A. Imteaj, U. Thakker, S. Wang, J. Li, and M. H. Amini, "A Survey on Federated Learning for Resource-Constrained IoT Devices," *IEEE Internet of Things Journal*, vol. 9, no. 1, pp. 1–24, Jan. 2022, https://doi.org/10.1109/JIOT.2021.3095077.

[2] H. Elahi, K. Munir, M. Eugeni, S. Atek, and P. Gaudenzi, "Energy Harvesting towards Self-Powered IoT Devices," *Energies*, vol. 13, no. 21, Nov. 2020, Art. no. 5528, https://doi.org/10.3390/en13215528.

[3] T. Cao Pham, H. Bich Vo, and N. Quang Tran, "A Design of Greenhouse Monitoring System Based on Low-Cost Mesh Wi-Fi Wireless Sensor Network," in *2021 IEEE International IOT, Electronics and Mechatronics Conference*, Toronto, Canada, 2021, pp. 1–6, https://doi.org/10.1109/IEMTRONICS52119.2021.9422655.

[4] Ö. Şeker and G. Dalkiliç, "Implementation and Performance Analysis of a Multi-Protocol Gateway," in *2022 Innovations in Intelligent Systems and Applications Conference*, Antalya, Turkey, 2022, pp. 1–6, https://doi.org/10.1109/ASYU56188.2022.9925244.

[5] L. M. Broell, C. Hanshans, D. Kimmerle, L. M. Broell, C. Hanshans, and D. Kimmerle, "IoT on an ESP32: Optimization Methods Regarding Battery Life and Write Speed to an SD-Card," in *Edge Computing - Technology, Management and Integration*, S. Goundar, Ed. London, United Kingdom: IntechOpen, 2023, ch. 7, https://doi.org/10.5772/intechopen.110752.

[6] M. E. Yuksel, "Power Consumption Analysis of a Wi-Fi-based IoT Device," *Electrica*, vol. 20, no. 1, pp. 62–70, Feb. 2020, https://doi.org/10.5152/electrica.2020.19081.

[7] A. C. Gheorghe, E. Stan, and I. Udroiu, "Electricity Consumption Measurement System Using ESP32," *The Scientific Bulletin of Electrical Engineering Faculty*, vol. 21, no. 2, pp. 23–26, Feb. 2022, https://doi.org/10.2478/sbeef-2021-0017.

[8] K. V. Sowmya, T. N. V. S. Pavan, V. Teju, G. S. D. Prasad, and P. V. Krishna, "IoT based Energy Monitoring in Smart Home using Mobile Application," in *2021 3rd International Conference on Advances in Computing, Communication Control and Networking*, Greater Noida, India, 2021, pp. 771–776, https://doi.org/10.1109/ICAC3N53548.2021.9725622.

[9] W. He and M. T. Iqbal, "Power Consumption Minimization of a Low-Cost IoT Data Logger for Photovoltaic System," *Journal of Electronics and Electrical Engineering*, vol. 2, no. 2, pp. 241–261, Dec. 2023, https://doi.org/10.37256/jeee.2220233795.

[10] P. Mayer, M. Magno, and L. Benini, "Smart Power Unit—mW-to-nW Power Management and Control for Self-Sustainable IoT Devices," *IEEE Transactions on Power Electronics*, vol. 36, no. 5, pp. 5700–5710, May 2021, https://doi.org/10.1109/TPEL.2020.3031697.

[11] G. Callebaut, G. Leenders, J. Van Mulders, G. Ottoy, L. De Strycker, and L. Van der Perre, "The Art of Designing Remote IoT Devices—Technologies and Strategies for a Long Battery Life," *Sensors*, vol. 21, no. 3, Feb. 2021, Art. no. 913, https://doi.org/10.3390/s21030913.

[12] A. Fanariotis, T. Orphanoudakis, and V. Fotopoulos, "Reducing the Power Consumption of Edge Devices Supporting Ambient Intelligence Applications," *Information*, vol. 15, no. 3, Mar. 2024, Art. no. 161, https://doi.org/10.3390/info15030161.