Heliyon 10 (2024) e40415

Contents lists available at ScienceDirect

Heliyon

journal homepage: www.cell.com/heliyon

Research article

Comprehensive electrical models for a wireless sensor network device

Bernardo Yaser León Ávila [a,b,*], Carlos Alberto García Vázquez [a,c], Osmel Pérez Baluja [a,c], Daniel Tudor Cotfas [a], Petru Adrian Cotfas [a]

[a] *Transilvania University of Brașov, Department of Electronics and Computers, Faculty of Electrical Engineering and Computer Science, Brasov, 500036, Romania*

[b] *University of Sancti Spiritus "José Martí Pérez", Department of Computer Engineering, Faculty of Business and Technical Sciences, Comandante Manuel Fajardo s/n, Olivos 1, Sancti Spíritus, 60100, Cuba*

[c] *Technological University of Havana, Department of Automation and Computing, Faculty of Automation and Biomedical Engineering, Street 114 No. 11901 between Ciclovía and Rotonda, Havana, 19390, Cuba*

A R T I C L E I N F O

Keywords:
Wireless sensor network node
Electric model
SPICE model

A B S T R A C T

This paper proposes a novel circuit model of a Wireless Sensor Network (WSN) device. The model is designed to accurately represent the behavior of a WSN device operating in a duty cycle, capturing the essential characteristics of its components, including the microcontroller, sensor, transceiver, and power supply. The proposed model incorporates static and dynamic power consumption aspects, reflecting the energy usage patterns during various operational states such as sensing, processing, and communication. The model's accuracy is validated through a comparative analysis with a LTspice® implementation and a real mote implemented for this study. Simulation results closely match the empirical data, demonstrating the model's effectiveness in predicting the mote's behavior. This approach provides a valuable tool for optimizing power consumption and extending the lifetime of WSN deployments within a simulation environment, offering significant insights for designers and researchers in this field.

1. Introduction

Wireless Sensor Networks (WSNs) consist of spatially distributed autonomous devices known as motes [1] designed to monitor physical or environmental conditions such as temperature, humidity, vibration, or pressure [2]. The ability to collect and transmit data wirelessly makes WSNs invaluable in scenarios where wired connections are impractical, enabling the deployment of sensors in remote or inaccessible locations [3]. For that reason, the WSNs have garnered significant attention due to their wide applications in environmental monitoring, healthcare, industrial automation, and smart cities [4].

The lack of human intervention in remote environments poses a significant challenge in terms of powering the various elements that comprise a WSN. This results in various mechanisms for Energy Management Strategy (EMS), encompassing different degrees of complexity, which may include energy harvesting [5].

It is possible to separate the energy management strategies from two different entities, one that is dedicated to ensuring the energy

* Corresponding author. Transilvania University of Brașov, Department of Electronics and Computers, Faculty of Electrical Engineering and Computer Science, Brasov, 500036, Romania.

E-mail address: bernardo.coca@unitbv.ro (B.Y. León Ávila).

https://doi.org/10.1016/j.heliyon.2024.e40415
Received 8 August 2024; Received in revised form 13 November 2024; Accepted 13 November 2024
Available online 15 November 2024
2405-8440/

Acronym list

ADC	Analog/Digital Converter
EMS	Energy Management Strategy
IC	Integrate Circuit
MCU	Microcontroller Unit
PCB	Printed Circuit Board
PMS	Power Management Systems
SQW	Sum of Square Waves
SMD	Surface-Mount Device
SPICE	Simulation Program with Integrated Circuit Emphasis
WSN	Wireless Sensor Network
WDT	Watchdog Timer

needs of the mote [6] known as the Power Management System (PMS), and the mechanisms that the mote establishes from its programming to save energy, that is, only work when it is strictly necessary by activating only the hardware resources involved in the specific operation and according to energy availability [7]. To achieve an accurate understanding of the energy consumption dynamics and to optimize the design of PMS in WSN devices, it is essential to develop comprehensive electrical models that can be incorporated into circuit simulations.

Electrical models are essential tools for understanding, predicting, and optimizing the behavior of electronic circuits and systems [8]. Accurate electrical modeling of WSN motes is crucial for several reasons [9]. Firstly, it allows for precise prediction of power consumption, which is essential for the design of energy-efficient WSNs. Given that devices often rely on battery power, extending the battery life is a primary concern. Secondly, electrical models enable designers to simulate and analyze the behavior of the motes under different operating conditions, facilitating the optimization of both hardware and software components [10]. An effective electrical model helps to understand the trade-offs between performance and energy efficiency, guiding the development of more sustainable WSN solutions.

Once the model has been developed, it should be subjected to a series of simulations in order to ascertain its efficacy and identify any potential issues. SPICE (from acronyms Simulation Program with Integrated Circuit Emphasis) is a widely adopted framework for circuit simulation that provides a standard method for modeling and analyzing the behavior of electronic circuits [11]. SPICE employs a netlist format that adheres to a specific syntax for the description of electronic circuits from component primitives. Practical models are then created to represent real, practical versions of these electronic components. These practical models may take the form of discrete circuit elements, such as real diodes or transistors, integrated circuits like the 555 timer, or even entire circuits viewed as subcircuits. LTspice® was chosen for its high performance and ease of use. According to Mohindru and Mohindru, 2021 [12], it is a powerful circuit simulator with a wide-ranging library of component models, making it ideal for modeling, analyzing, and displaying analog circuits.

Modeling the electrical behavior of WSN devices presents several challenges. These nodes operate in various modes, including active, sleep, and idle states, each with distinct power consumption profiles. Furthermore, the interactions between the microcontroller, sensors, transceiver, and power supply are complex and require detailed characterization. Accurately capturing these dynamics is essential for developing reliable models. The variability in power consumption due to environmental factors and operational conditions adds another layer of complexity to the modeling process.

Most of the mote's activity is dedicated to network communication, with the radio interface serving as the physical layer of the network stack and representing the largest portion of power consumption. The activity of this interface depends on the network's particularities such as protocol stack, topology, role, logical and physical positions, interference, the length of the data transmitted, etc. [13]. This may account for the fact that studies obtained from a non-exhaustive search on the topic present energy consumption models based on network simulation software such as Ns-3, Castalia, TOSSIM, and OPNET [14], and COOJA [15,16], or using a dedicated software developed by the author of the search [17]. This software offers accurate models that can be simulated with the necessary realism of the mote's energetic behavior and subsequently exported in a file that can be incorporated into the SPICE simulation. However, this procedure necessitates a higher computational cost, translating into more complex and time-consuming simulations involving several steps in more than one software. Moreover, energy consumption can vary from one device to another depending on the factors described above.

Other studies consider power consumption as the average of the different states like idleness, sensing and logging data, and transmitting and receiving data. In this case, the simulation of the isolated node considers the average network behavior in a particular WSN device. Examples include Bakr and Lilien, 2011 [18], Ke et al., 2015 [19], Ghosh and Unnikrishnan, 2017 [20] and Mayer et al., 2022 [21] using MATLAB to obtain a power consumption average consider a simple alternation between states, Kellner et al., 2008 [22] describing a mote as a finite-state machine, Kempa, 2019 [23] proposed a threshold-controlled vacation policy, Blondia, 2021 [24] present a generic queueing model resulting in a discrete-time Markov Chain, and Li et al., 2014 [25] use Queuing Petri Net (QPN). This approximation leads to a simple model that is cleaner, less demanding, faster, and accurate enough. Nevertheless, all these proposals require an algorithmic description through some programming language, contrary to the SPICE work philosophy. In this sense, no study has been identified that provides detailed circuit-level models that can be directly utilized in SPICE simulations.

The **objective** of this paper is to develop a comprehensive electrical model described of a WSN device. The model is designed to replicate the behavior of a WSN device operating in a duty cycle, accounting for both static and dynamic power consumption. By providing a detailed representation of the mote's components and their interactions, the model becomes a valuable tool for predicting power consumption and optimizing power electronics designs. The LTspice® environment is chosen as SPICE environment for its robust simulation capabilities and widespread use in electronic circuit design.

This paper is organized as follows: In Section 2 an overview of WSN node components and their electrical characteristics is provided. Also, this section details the development of the LTspice® circuit model and the physical mote itself. Section 3 presents the simulation setup and results comparing these results with empirical data from the real mote. Finally, the paper's Conclusion is presented, including suggestions for future work.

2. Materials and methods

This section outlines the materials and methodologies employed to develop and validate the model for a WSN mote. The model aims to accurately represent the mote's behavior, focusing on energy consumption and state transitions. The process involved selecting appropriate primitive components, designing the circuit in the LTspice® environment, and implementing a real mote to compare the simulated results performance. The following subsections detail the hardware configuration, circuit design, and simulation parameters.

2.1. Mote consumption model

From the point of view of this analysis, the WSN device represents only a load for the PMS as Fig. 1 shows. The highest power consumption of a WSN node could be estimated by the simple sum of the worst values given by the manufacturers of the components that it comprises, but this would be unrealistic.

The electrical characteristics of the key components of a WSN mote are critical for understanding and modeling its power consumption and overall performance. The mote's behavior is split into various operational states [19] with different components influencing it through its unique properties. The main components are [26].

- **Microcontroller Unit (MCU):** Its current consumption varies significantly depending on the operational mode. In active mode, where the MCU is executing instructions, the current consumption ranges from a few milliamperes (mA) to tens of milliamperes. In contrast, during sleep mode, the MCU's current consumption drops substantially to microamperes (μA) or nanoamperes (nA). The clock frequency, which determines the speed of operation, also affects the power consumption, with higher frequencies leading to increased current draw. Additionally, using integrated peripherals such as ADCs, serial communications, and timers impacts the overall power consumption.
- **Sensors**: Their electrical characteristics vary depending on the type and application. Their current consumption also varies, often within the microampere to milliampere range. During active sensing, sensors draw more current than in idle states when is significantly lower.
- **Radio interface (transceiver):** The current consumption varies based on the operational state. In transmit mode the current consumption is high, often reaching tens of milliamperes. In receive mode, the current draw is slightly lower but still significant. When in idle or standby mode, the transceiver consumes much less current.

Depending on the application, a WSN mote may include additional peripherals such as memory modules or real-time clocks (RTC) [27]. Non-volatile memory modules, are used for data storage and have current consumption dependent on read-write operations [28].

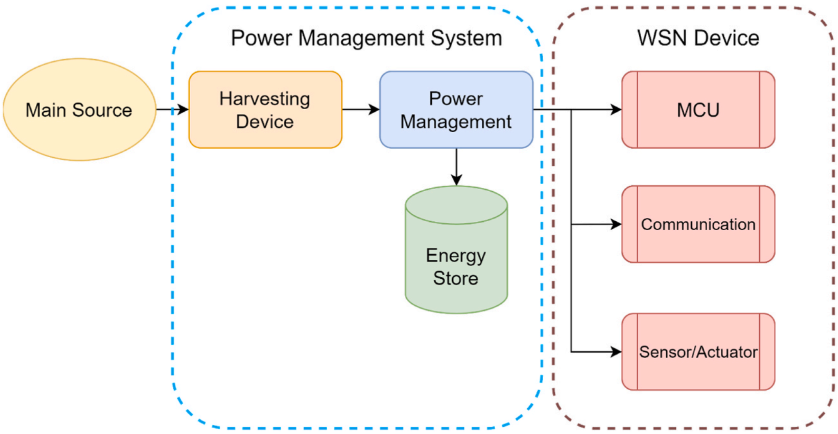

Fig. 1. Overall representation of mote including energy harvesting system.

Actuators, which perform actions based on sensor data, can vary widely in their current consumption based on their function and usage.

Therefore, the current demanded by a mote will include an offset from all continuously operating elements, including the MCU in its lowest consumption mode. Furthermore, there will be different plateaus at regular intervals, contingent on the device that is active and its specific task. The resulting graph, which will be discussed in more detail in the following sections, is called a consumption profile. This graph reflects the mote's behavior and is the objective of the simulation. Depending on the point from which the consumption profiles are extracted, DC/DC regulators or protections could be included in the model, which will be reflected as an offset of the global profile.

2.2. Electrical model construction

In developing the model, existing approaches that posit a periodic change in the mote's state contingent upon the occurrence of specific events have been built upon. Since there are multiple and complex mechanisms for energy management that modulate the behavior of a mote in its interaction with the environment, the triggers are simplified as time-fixed events with a specific frequency. Each state is associated with a specific load within the system. This means that the simulated mote will exhibit a strict periodic behavior, mirroring the real mote's potential behavior under controlled conditions. In this case, a device that exhibits adaptive behavior will align with the model only under one of these controlled conditions.

In this way, each component of the mote contributes as a load that periodically switches between two states. Each load will be represented as a *current source* because a resistor will vary the consumption depending on the applied voltage. Each current source represents the states of each element as a pulse function. The base result is a function that is the sum of all current functions expected as a sum of square waves (SQW), with amplitude A_i, frequency f_i and phase ϕ_i, and could be expressed as:

$$y(t) = \sum_{i=0}^{N} SQW(A_i, f_i, \phi_i) \tag{1}$$

The model is completed with a source representing the mote in the lowest consumption mode, an aleatory signal simulating noise and an RLC network. Fig. 2 presents the proposed electrical model designed directly in LTspice® environment.

In the model, shown in Fig. 2, the current source uses a pulse function where the parameters passed to the function have the form:

$$PULSE\left(I_{off}, I_{on}, T_{delay}, T_{rise}, T_{fall}, T_{on}, T_{period}\right)$$

Where I_{on} and I_{off} are the consumption states, T_{on} the time in a high consumption state into a period T_{period}, T_{delay} define the phase as a delay and T_{rise} and T_{fall} represent the times of state change from off to on and vice versa. The number of current sources and their values will depend in each case on the model to be simulated. In this case, the values presented in Fig. 2 will be discussed in subsequent sections.

To simplify, the MCU only switches between the sleeping and running states. Assuming that it is only running when having an in/out operation, the power consumption of the run state is added to the element that operates, in this case, the radio interface or the sensor. Each state is represented as a pulse current source; receiving and transmitting have different states, and then, they are represented individually.

The proposed model aims to replicate the behavior of the real mote. Achieving this in the SPICE environment is challenging because network conditions influence real behavior. The model assumes ideal network conditions, without packet loss or retransmissions, and uniform frame lengths. Despite these simplifications, the model still offers a readable and accurate representation of circuit behavior. A

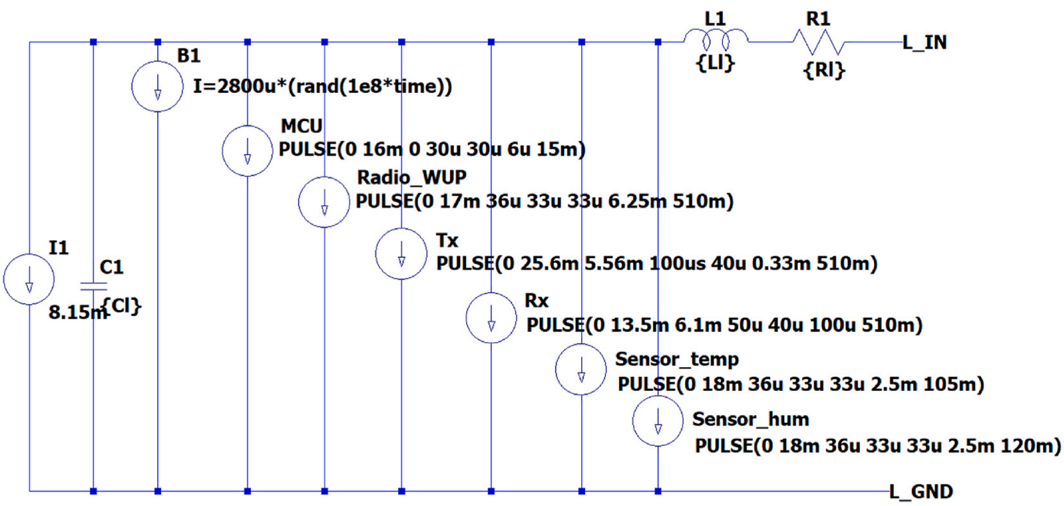

Fig. 2. WSN electrical node model represented in LTspice® environment.

detailed comparison between the model and the real WSN node will be presented to validate the model's accuracy.

It uses SPICE's own netlist syntax and can be directly ported to other SPICE-based simulators such as LTspice, HSPICE, Kicad, PSpice, Ngspice, Proteus, Qucs, Multisim, etc. In the event of the model being reproduced for use with alternative non-SPICE simulators, such as Simulink in MATLAB [29], it is essential that the reproduction is carried out in accordance with the specifications and underlying philosophy of the selected software. One advantage of SPICE-based simulation software is that it often includes models of the actual integrated circuits provided by the manufacturers.

2.3. The development of the mote

In order to validate the model, a mote prototype has been constructed in a simple manner, combining well-known commercial components and modules to reproduce a generic WSN device as described in Fig. 1.

This simplicity has the additional benefit of facilitating the testing tasks and reproduction of the experiment. The result is illustrated in Fig. 4 as a schematic and in Fig. 3 as a circuit mounted in a PCB.

An Arduino Nano is used as the control unit. The board includes as MCU an ATMega328p microcontroller, a CH340 as a USB interface, and a 16 MHz resonator as the clock frequency. The sensors and radio interface are a BMP280 and an nRF24L01 respectively. A receiver is used in the same structure and elements but without the sensor as is shown in Fig. 4. In the case of the receiver, the nRF24L01 module incorporating an RF amplifier and antenna is used.

The mote is powered at 5V by a regulated laboratory power supply across a 1Ω shunt resistor. The shunt resistor is a surface-mount device (SMD) component situated on the reverse side of the printed circuit board (PCB) in comparison to the illustration in Fig. 4, the image of the shunt resistor is displayed over the photograph, and a circle indicates the position on the opposite side of the PCB.

The measurements are conducted using the NI-Elvis III [30] platform oscilloscope on the referred shunt resistor, allowing the Y-axis to be interpreted interchangeably as voltage or current.

Ultimately, the firmware determines the behavior of the mote. As part of this research was developed idOS [31], a cooperative scheduler with protothreads [32] as a process model has been developed using the Arduino framework. The firmware, which implements the typical behavior of a mote, also incorporates timers, an energy management system based on system ticks, and a comprehensive network stack based on uIP [33], with CoAP [34] implemented as the application layer protocol for WSNs.

In this system, the scheduler runs all the tasks in the queue and, upon completion, puts the MCU to sleep. The Watchdog Timer (WDT) periodically wakes up the MCU through the power management module. Once awake, the system checks the timers, and the

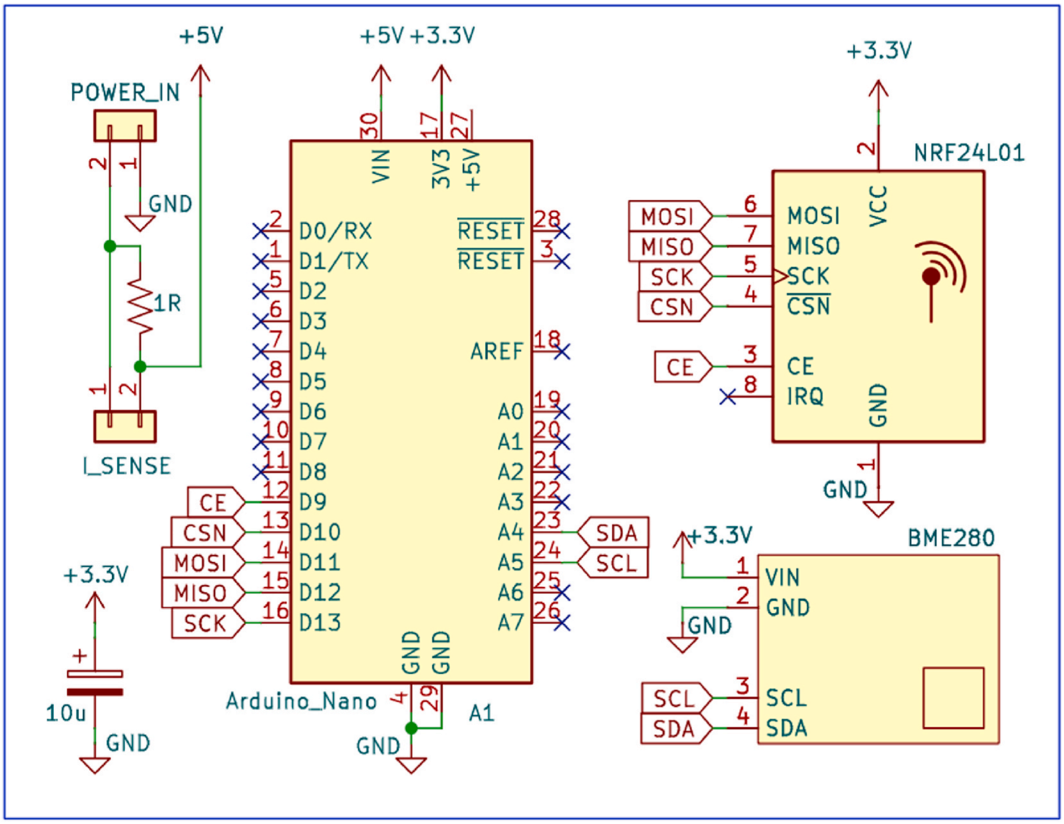

Fig. 3. Schematic of a mote.

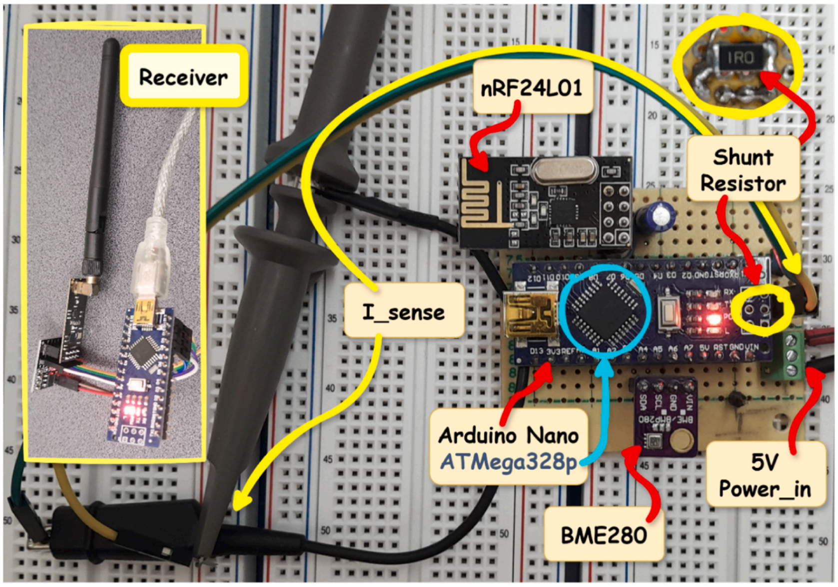

Fig. 4. The mote prototype. At the left, inside the yellow rectangle, the receiver is powered by a USB cable. On the right, the shunt resistor, located below the PBC, is indicated by the yellow circle. (For interpretation of the references to colour in this figure legend, the reader is referred to the Web version of this article.)

network, and runs the scheduler. This repeating cycle represents the system tick. Fig. 5 shows the energy profile of a mote running the system. The graphs provide a detailed depiction of the behavior of each of the states described above. As mentioned, the measurement is taken from a 1Ω shunt resistor (R). The Y-axis in Fig. 5 shows the voltage values (V), which can be interpreted as current (I) values following Ohm's law, where $I = V/R$, if $R = 1$ then $I = V$.

Fig. 5 b) provides a detailed illustration of the most complex event, the activity of the radio interface that is constituted of four distinct sub-events. In this particular case, the starting process of radio interface exhibits an irrelevant current consumption in comparison to the MCU. In order to simplify the model, the time taken for the MCU and the time taken for the radio interface to start up will be considered as a single sub-event. Their corresponding data reflected in Table 1.

The approach employed in this investigation to construct the hardware ensures expeditious and dependable replication of the motes; nevertheless, it also entails the addition of a multitude of unnecessary components that consume energy, rendering the motes unsuitable for practical applications. This additional consumption is observed as an offset consumption level of approximately 9 mA in the energy profile illustrated in Fig. 5 a).

3. Results and discussions

According to the ATMega328p [35], nRF24L01 [36] and BME280 [37] datasheets, the maximum expected consumption respectively is 66 μA, 0.9 μA, and 0.3 μA in power-down mode, and this value will be used in the simulation model. An offset of 9 mA was added as was described before.

In the experiment, the mote was configured to run the schedule every 15 ms. Temperature and humidity were measured approximately every 100 and 110 ms, respectively, and transmitted every half second. Using the consumption profile shown in Fig. 4, the real values of the node's behavior pattern were extracted to pass them to the model. These values are found in Table 1. Note that, from an energy perspective, the activity of the radio interface comprehends the combined energy consumption of radio wake-up, transmission, and reception phases.

Because between each tick the OS is asleep so it cannot do anything, each T_e will always occur at a time multiple of T (tick), where T_e represent the period of occurrence of each of the different events that force change state from *off* to *on*. Typically, events are scheduled (setting the event's particular timer) to occur within a given time T_e. Thus, the event will occur with a delay of T_{delay} according to the following equation.

$$T_{delay} = T - \left(T_e - \left\lfloor {}^{T_e}\!/_{T} \right\rfloor * T\right) \tag{2}$$

Where the real execution period of T_{period} is expressed as:

$$T_{period} = T_e + T_{delay} \tag{3}$$

Fig. 5. Mote consumption profile; a) is a larger view showing the system ticks, as well as the activity of the sensors and the radio interface; b) is a detailed view of the electrical behavior of the radio interface.

Table 1
Consumption values by states extracted from the mote profile.

State	I (mA)	Duty cycle	
		On (msec)	Off (msec)
CPU Idle	16	0.006	15
Sensing temperature	18	2.5	100
Sensing humidity	18	2.5	100
Radio wakes up	17	5.45	500
Radio Tx	43	0.33	500
ACK Rx response	30.5	0.1	505.1
Noise fluctuation	2.3	–	–

$$T_{period} = T\left(1 + \left\lfloor {}^{T_e}\!/_{T} \right\rfloor\right) \tag{4}$$

Based on these considerations, the model can be recalibrated to reflect the mote's behavior in a more energy-restrictive environment. The SPICE's netlist for simulation was updated according to Table 1 values and the delay time obtained by applying Equation (4) for each component. In this scenario, the Microcontroller Unit (MCU) duty cycle, the radio interface activity period, and the sensing period of both variables are adjusted to 1 s, 60 s, and 29 s, respectively.

Fig. 6 demonstrates the model's accuracy in replicating the behavior of the real mote. The comparison illustrates that the simulated data closely matches the measurements obtained from the actual device. This alignment indicates that the model successfully captures the essential dynamics and characteristics of the mote's operation.

The figure highlights key aspects such as the timing of sensor readings, transmission intervals, and overall power consumption patterns. The close correspondence between the model and real measurements validates the assumptions and simplifications made in the model, confirming its reliability for predicting the mote's performance under various conditions. This validation is crucial for ensuring that the model can be used to optimize mote configurations and improve energy efficiency in practical applications.

In order to validate the behavior of the model from the point of view of energy consumption, the energy consumption of the real machine will be measured and compared with the simulation model. Fig. 6 shows the consumption progress in 7 s of simulation and measurement on the real prototype. It is noteworthy how the model and the prototype have practically the same behavior.

To obtain the values presented in the graph in Fig. 7, the energy was calculated every 100 ms. In the case of the simulation, it was obtained from the integration statement in 70 intervals. The following is a representation of the statements used in LTspice:

$$.measure\ tran\ E\_01\ INTEGRAL\ Ix(U1:L\_IN)*V(n001)\ FROM\ 0\ TO\ 0.1$$

$$.measure\ tran\ E\_02\ INTEGRAL\ Ix(U1:L\_IN)*V(n001)\ FROM\ 0\ TO\ 0.2$$

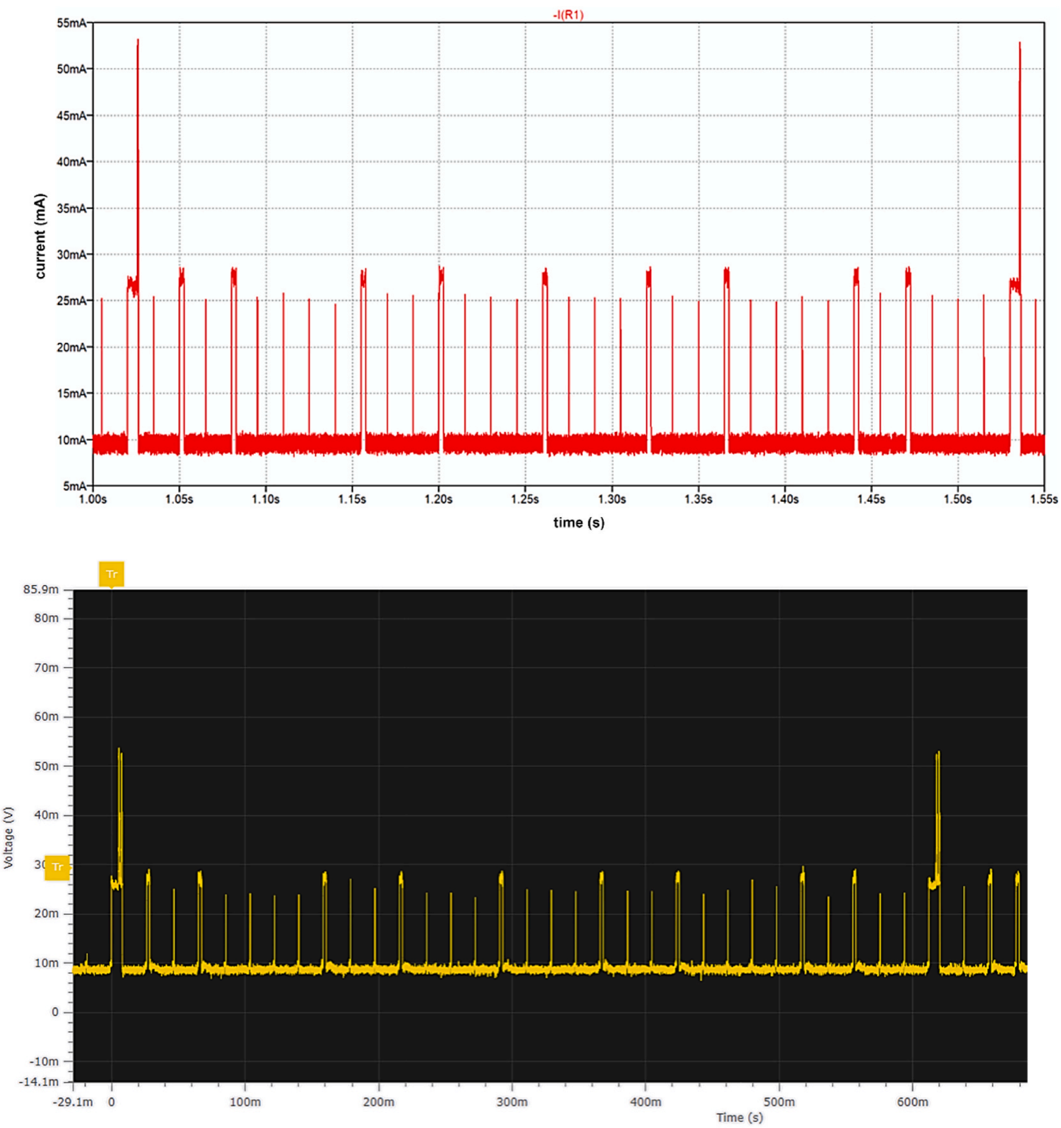

Fig. 6. The simulated node (graphs with red lines and white background) vs the real node (graphs with yellow lines and black background). The first two pairs of graphs show the entire cycle between radio interface activities, while the second two pairs of graphs provide a zoomed-in view of the transceiver activity. (For interpretation of the references to colour in this figure legend, the reader is referred to the Web version of this article.)

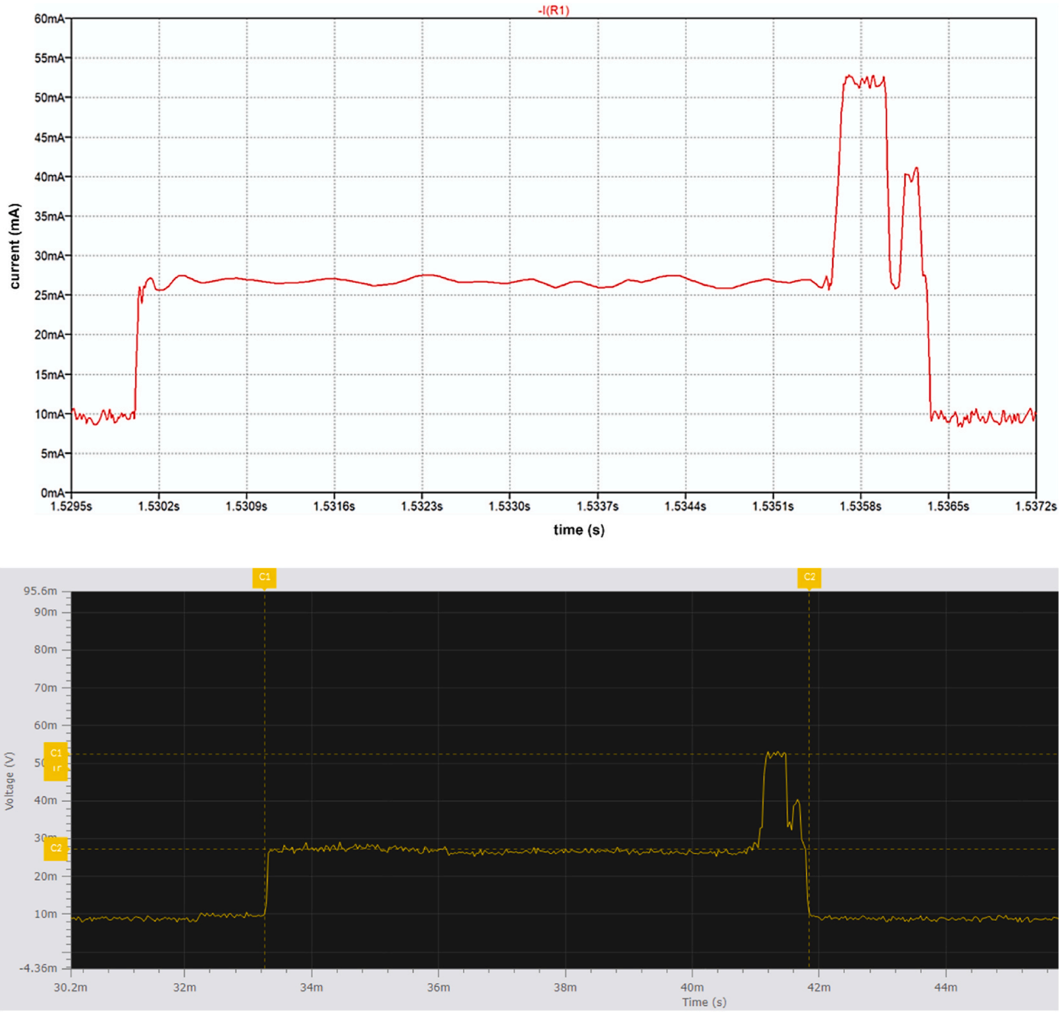

Fig. 6. (*continued*).

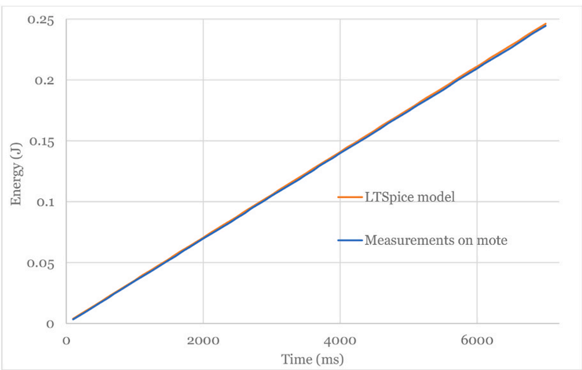

Fig. 7. Progress of energy consumption measured on real mote (blue line) versus model running in LTspice®. (For interpretation of the references to colour in this figure legend, the reader is referred to the Web version of this article.)

…

$.measure\ tran\ E\_70\ INTEGRAL\ Ix(U1:L\_IN)*V(n001)\ FROM\ 0\ TO\ 7.0$

The energy consumed by the mote was calculated from the energy profile measured by Elvis III's Datalogger tool sampling at 100 KHz for 7 s on the shunt resistance described above. Since the shunt resistance is 1 Ω (verified) the voltage value measured by the datalogger can be directly interpreted as the value of the electric current. The samples were processed with Microsoft Excel, calculating the energy as a summation of the instant power of each sample from 0 to each point multiple of 10000, as shown as follows:

$$Energy = \sum_{n=1}^{N} (V \times i_n) \times \Delta t \tag{5}$$

Where the P is calculated by multiplying the electric current by the supply voltage V that powers the mote. Because the calculation is performed at intervals of every 100 ms, and values of N are taken at each multiple of 10,000 until 7 s have elapsed and, for a sampling rate of 100 KS/se, the value of $\Delta t = 10^{-5}$ *sec*.

The mote will exhibit strict periodic behavior, reflecting the real mote's potential performance under controlled conditions. In complex or harsh environments where a device must adapt its behavior (e.g., by increasing or decreasing the transmission frequency), this model may not accurately follow these adaptive changes. Instead, the model can only describe the behavior of one particular state. For instance, the worst-case scenario involves the highest transmission frequency, which increases the duration of the Tx and Rx states. Similarly, in scenarios involving retransmissions, the number of Tx cycles must be increased, depending on the network protocol behavior.

Each mote case must be built based on its specific electrical behavior, derived from its consumption profile as described in this paper. SPICE models cannot fully capture the complex and dynamic nature of network scenarios. Therefore, it is necessary to simplify and approximate the electrical behavior to provide references for designing and testing auxiliary circuits, such as PMS, in simulations.

4. Conclusions

This paper presented a comprehensive circuit model of a Wireless Sensor Network (WSN) mote designed to replicate the behavior of a real WSN device operating in a duty cycle. By incorporating the electrical characteristics of key components such as the microcontroller, sensors, transceiver, and power supply, the model provides an accurate representation of the mote's power consumption and performance. The model's validity was demonstrated through a comparative analysis between simulation, utilizing the LTspice® environment, and a real mote specifically designed for this study. From a network perspective, this model may lack precision and may not yield useful insights. However, from a circuit design and simulation standpoint, particularly within environments like SPICE, it serves as an invaluable tool for testing the host's supporting electronics, including power management systems and energy harvesting components.

Despite the assumptions of ideal network conditions, the simulation results closely matched the empirical data, proving the model's effectiveness in predicting the mote's behavior. This validated model provides a valuable tool in various circuit simulation environments, optimizing power consumption, facilitating the design of more energy-efficient WSN nodes, and extending the operational lifetime of WSN deployments.

Future work exploring the integration of energy harvesting techniques and advanced power management strategies could provide deeper insights into the sustainable deployment of WSNs.

CRediT authorship contribution statement

Bernardo Yaser León Ávila: Writing – original draft, Validation, Software, Methodology, Investigation, Formal analysis, Conceptualization. **Carlos Alberto García Vázquez:** Writing – review & editing, Validation, Investigation. **Osmel Pérez Baluja:** Writing – review & editing, Visualization. **Daniel Tudor Cotfas:** Writing – review & editing, Validation, Supervision, Resources. **Petru Adrian Cotfas:** Writing – review & editing, Validation, Supervision, Resources.

Data availability

The data supporting the results of this study are available from the corresponding author upon reasonable request. The SPICE code of the devices simulated in further work from the described, proposed models, and the firmware for the mote has been published under the GNU license on GitHub.

Ethics declaration

Review and/or approval by an ethics committee as well as informed consent was not required for this study because this article did not involve any direct experimentation/studies on living beings.

Declaration of generative AI and AI-assisted technologies in the writing process

During the preparation of this work the authors used ChatGPT in order to improve the readability and language of the manuscript.

After using this service, the authors reviewed and edited the content as needed and takes full responsibility for the content of the published article.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] L.K. Ketshabetswe, A.M. Zungeru, M. Mangwala, J.M. Chuma, B. Sigweni, Communication protocols for wireless sensor networks: a survey and comparison, Heliyon 5 (5) (May 2019) e01591, https://doi.org/10.1016/j.heliyon.2019.e01591.
[2] K. Olcay, E. Taparci, M.O. Akmandor, B. Kabakulak, B. Sarioglu, Y.D. Gokdel, Modeling and implementation of an adaptive wireless sensor network for low power IoT applications. Presented at the 2023 8th International Conference on Smart and Sustainable Technologies, SpliTech 2023, 2023, https://doi.org/10.23919/SpliTech58164.2023.10193646.
[3] S. Mohan, S. Panda, Multi-factor optimized mobile sink data collection framework for hybrid WSN-LTE assisted IoT network, Heliyon 10 (5) (Mar. 2024) e25998, https://doi.org/10.1016/j.heliyon.2024.e25998.
[4] K.C. Okafor, O.M. Longe, Smart deployment of IoT-TelosB service care StreamRobot using software-defined reliability optimisation design, Heliyon 8 (6) (Jun. 2022) e09634, https://doi.org/10.1016/j.heliyon.2022.e09634.
[5] X. Liu, et al., Sponge Supercapacitor rule-based energy management strategy for wireless sensor nodes optimized by using dynamic programing algorithm, Energy 239 (Jan. 2022) 122368, https://doi.org/10.1016/j.energy.2021.122368.
[6] N. Qi, Y. Yin, K. Dai, C. Wu, X. Wang, Z. You, Comprehensive optimized hybrid energy storage system for long-life solar-powered wireless sensor network nodes, Appl. Energy 290 (May 2021) 116780, https://doi.org/10.1016/j.apenergy.2021.116780.
[7] J. Fu, X. Ma, H. Yu, K. Dai, Distributed energy-efficient wireless sensing and information fusion via event-driven and state-rank activation, Wirel. Netw. 30 (4) (May 2024) 2697–2711, https://doi.org/10.1007/s11276-024-03691-8.
[8] R. Aalloul, A. Elaissaoui, A. Harkani, R. Adhiri, M. Benlattar, A simulation and modeling approach of coupled thermal and electrical behavior of PV panels using the artificial hummingbird algorithm and two-dimensional finite difference-based model, Heliyon 10 (6) (Mar. 2024) e27244, https://doi.org/10.1016/j.heliyon.2024.e27244.
[9] M.B. Marjanovic, A. Stojkovic, A. Prijic, D. Dankovic, Z. Prijic, Spatial SPICE model of a wireless sensor network node based on a thermoelectric generator, Facta Univ. Ser. Electron. Energ. 35 (4) (Dec. 2022), 4.
[10] V. Ngunzi, F. Njoka, R. Kinyua, Modeling, simulation and performance evaluation of a PVT system for the Kenyan manufacturing sector, Heliyon 9 (8) (Aug. 2023) e18823, https://doi.org/10.1016/j.heliyon.2023.e18823.
[11] L.W. Nagel, D.O. Pederson, SPICE (Simulation Program with Integrated Circuit Emphasis), EECS Department, University of California, Berkeley, UCB/ERL M382, 1973 [Online]. Available: http://www2.eecs.berkeley.edu/Pubs/TechRpts/1973/22871.html.
[12] P. Mohindru, P. Mohindru, Electronic Circuit Analysis Using LTSpice XVII Simulator: A Practical Guide for Beginners, CRC Press, Boca Raton, 2021, https://doi.org/10.1201/9781003199489.
[13] S. Srinivasan, T.K. Ramesh, R. Paccapeli, L. Fanucci, Industrial functional safety assessment for WSN using QoS metrics, Heliyon 8 (11) (Nov. 2022) e11255, https://doi.org/10.1016/j.heliyon.2022.e11255.
[14] J. Helkey, L. Holder, B. Shirazi, Comparison of simulators for assessing the ability to sustain wireless sensor networks using dynamic network reconfiguration, Sustain. Comput. Inform. Syst. 9 (Mar. 2016) 1–7, https://doi.org/10.1016/j.suscom.2016.01.003.
[15] W.M. Dahmane, S. Ouchani, H. Bouarfa, Towards a reliable smart city through formal verification and network analysis, Comput. Commun. 180 (Dec. 2021) 171–187, https://doi.org/10.1016/j.comcom.2021.09.006.
[16] J.A. Ternero, V. Mayor, R. Estepa, A. Estepa, G. Madinabeitia, Minimizing energy consumption in 802.15.4 IoT devices with multilevel xRPL (MxRPL), Internet Things 24 (Dec. 2023) 100981, https://doi.org/10.1016/j.iot.2023.100981.
[17] A.M.K. Abdulzahra, A.K.M. Al-Qurabat, S.A. Abdulzahra, Optimizing energy consumption in WSN-based IoT using unequal clustering and sleep scheduling methods, Internet Things 22 (Jul. 2023) 100765, https://doi.org/10.1016/j.iot.2023.100765.
[18] B.A. Bakr, L. Lilien, A quantitative comparison of energy consumption and WSN lifetime for LEACH and LEACH-SM, in: 2011 31st International Conference on Distributed Computing Systems Workshops, Jun. 2011, pp. 182–191, https://doi.org/10.1109/ICDCSW.2011.55.
[19] J.-F. Ke, W.-J. Chen, D.-C. Huang, Life extend approach based on priority Queue N strategy for wireless sensor network, in: 2015 11th International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness (QSHINE), Aug. 2015, pp. 272–279. Accessed: Jul. 29, 2024. [Online]. Available: https://ieeexplore.ieee.org/document/7332581.
[20] S. Ghosh, S. Unnikrishnan, Reduced power consumption in wireless sensor networks using queue based approach, in: 2017 International Conference on Advances in Computing, Communication and Control (ICAC3), Dec. 2017, pp. 1–5, https://doi.org/10.1109/ICAC3.2017.8318794.
[21] P. Mayer, M. Magno, L. Benini, Model-based design for self-sustainable sensor nodes, Energy Convers. Manag. 272 (Nov. 2022) 116335, https://doi.org/10.1016/j.enconman.2022.116335.
[22] S. Kellner, M. Pink, D. Meier, E.-O. BlaB, Towards a realistic energy model for wireless sensor networks, in: 2008 Fifth Annual Conference on Wireless on Demand Network Systems and Services, Jan. 2008, pp. 97–100, https://doi.org/10.1109/WONS.2008.4459362.
[23] W.M. Kempa, Analytical model of a wireless sensor network (WSN) node operation with a modified threshold-type energy saving mechanism, Sensors 19 (14) (Jan. 2019), https://doi.org/10.3390/s19143114, 14.
[24] C. Blondia, A queueing model for a wireless sensor node using energy harvesting, Telecommun. Syst. 77 (2) (Jun. 2021) 335–349, https://doi.org/10.1007/s11235-021-00758-1.
[25] J. Li, H.Y. Zhou, D.C. Zuo, K.M. Hou, H.P. Xie, P. Zhou, Energy consumption evaluation for wireless sensor network nodes based on queuing Petri Net, Int. J. Distrib. Sens. Netw. 10 (4) (Apr. 2014) 262848, https://doi.org/10.1155/2014/262848.
[26] A. Nayak, I. Stojmenovic, Wireless Sensor and Actuator Networks, John Wiley & Sons, New Jersey, EEUU, 2010.
[27] P. Loreti, A. Catini, M. De Luca, L. Bracciale, G. Gentile, C. Di Natale, The design of an energy harvesting wireless sensor node for tracking pink iguanas, SENSORS 19 (5) (Mar. 2019), https://doi.org/10.3390/s19050985.
[28] Y.J. Kim, et al., High-performance self-powered wireless sensor node driven by a flexible thermoelectric generator, Energy 162 (Nov. 2018) 526–533, https://doi.org/10.1016/j.energy.2018.08.064.
[29] C.A. García Vázquez, D.T. Cotfas, A.I. González Santos, P.A. Cotfas, B.Y. León Ávila, Reduction of electricity consumption in an AHU using mathematical modelling for controller tuning, Energy 293 (Apr. 2024) 130619, https://doi.org/10.1016/j.energy.2024.130619.
[30] A. Mayzel, A. Medvedev, V. Temkina, L. Pankova, E. Sivolenko, On the experience of using NI ELVIS III in remote laboratory practice during pandemic lockdown, in: E. Velichko, V. Kapralova, P. Karaseov, S. Zavjalov, P. Angueira, S. Andreev (Eds.), International Youth Conference on Electronics, Telecommunications and Information Technologies, Springer International Publishing, Cham, 2022, pp. 77–83, https://doi.org/10.1007/978-3-030-81119-8_8.
[31] B.Y.L. Ávila, et al., Protothread and Cooperative Multitasking Scheduler on the Arduino Framework, in: 2024 International Conference on Applied and Theoretical Electricity (ICATE), Oct. 2024, pp. 1–5, doi:10.1109/ICATE62934.2024.10749444.

[32] A. Dunkels, O. Schmidt, T. Voigt, M. Ali, Protothreads: simplifying event-driven programming of memory-constrained embedded systems, in: Proceedings of the 4th International Conference on Embedded Networked Sensor Systems, in SenSys '06, Association for Computing Machinery, New York, NY, USA, Oct. 2006, pp. 29–42, https://doi.org/10.1145/1182807.1182811.

[33] A. Dunkels, Full TCP/IP for 8-bit architectures, in: *Proceedings Of the 1st International Conference on Mobile Systems, Applications and Services*, in MobiSys '03, Association for Computing Machinery, New York, NY, USA, May 2003, pp. 85–98, https://doi.org/10.1145/1066116.1066118.

[34] D. Silva, L.I. Carvalho, J. Soares, R.C. Sofia, A performance analysis of internet of things networking protocols: evaluating MQTT, CoAP, OPC UA, Appl. Sci. 11 (11) (Jan. 2021), https://doi.org/10.3390/app11114879. Art. no. 11.

[35] Microchip Technology Inc., 'ATmega48A/PA/88A/PA/168A/PA/328/P', Microchip Technology Inc., 2018 [Online]. Available: https://www.microchip.com/en-us/product/atmega328pb. (Accessed 6 May 2024).

[36] Nordic Semiconductor, nRF24L01+ Single chip 2.GHz transceiver. Nordic Semiconductor, 2008.

[37] Bosch, 'BME280 Combined Humidity and Pressure Sensor', Bosch, 2018.