BeeGOns!: A Wireless Sensor Node for Fog Computing in Smart City Applications

Michael Vera-Panez, Kewin Cuadros-Claro, Manuel Castillo-Cara, and Luis Orozco-Barbosa, *Member, IEEE*

***Abstract*—The widespread deployment of sensors interconnected by wireless links and the management and exploitation of the data collected have given rise to the Internet of Things (IoT) concept. In this article, we undertake the design and implementation of a wireless multisensor platform following the fog computing paradigm. Our main contributions are the integration of various off-the-shelf sensors smartly packaged into an air-flow module and the evaluation of the communications services offered implemented on top of two low-power radio communications technologies. Our study is complemented by evaluating the communications services over a wired link. Our results show the superiority of LoRaWAN over ZigBee in terms of power consumption despite its slightly higher computational requirements and an estimation of the gap between the resource usage of the wired link and the two wireless radio technologies.**

***Index Terms*—Edge computing, embedded sensor networks, fog computing, LoRa, sensor nodes, smart city.**

I. Introduction

Over the past years, the development of wireless networks, mobile devices, and computing paradigms have made possible the deployment of a large number of information- and communication-assisted services [1]. Internet of Things (IoT) is being deployed to monitor or assist the operation of a wide variety of monitoring, control, and production processes [2]. Nowadays, behind city traffic control, food production tasks, IoT systems are being used to monitor the resources and effectiveness of the actions that manually or automatically are being taken in response to the information provided by sensors deployed across the monitored area [3].

Despite the promising results having been obtained up-to-date, there are still many issues to be addressed on the design and implementation of IoT-based systems capable of meeting the end-user application requirements [4], [5]. Besides the issues to be addressed to improve the operation of the underlying communication platforms, another major issue has to deal with the design of a robust and scalable data processing architecture [6]. Toward this end, the data processing architecture of IoT systems has moved from a cloud paradigm to a fog paradigm [7], [8].

Fog computing is an emerging paradigm of IoT technologies where processing, storage, and communications are geographically spread rather than centralized in a cloud infrastructure [6]. It is essential to develop an energy-efficient architecture integrating various processing, sensors, and communication technologies and services [3], [9]. Among the various elements of a Fog Computing architecture, the communications system must fulfill the following requirements: area coverage, scalability, reliability, and standardization [7], [10]. Moreover, the sensor nodes of a Fog Computing platform must exhibit autonomy in terms of energy and robustness, since many applications will require the deployment of such nodes in remote and harsh environment regions [8], [11]. As for energy consumption, Castillo-Cara et al. [8] and Ayala-Ruiz et al. [12] evaluated the energy consumption of ZigBee and LoRAWAN compliant mesh sensor networks, respectively. Other works have been mainly focused on enabling larger area coverage and better power efficiency [6].

Another major challenge relates to the diversity of data to be collected, some of which required the design of the sensor's packaging. In the context of a Smart City, air quality is a parameter of high interest to both authorities and citizens. Catlett et al. [13] described Array of Things (AoT), a Smart City project being carried out in Chicago. According to the authors, one of the main aims focused on developing a customizable hardware and software platform that can be upgraded by adding new sensing devices. Metia et al. [14] presented a Smart City platform specifically designed for air pollution monitoring. Their efforts have focused on protecting the various sensors and electronics against water and dust.

In this work, we go a step further. Besides developing a customizable platform, our platform packaging has been optimized to ensure a constant and adequate airflow, which allows the sensors to capture data representative of the air quality in an urban area. Therefore, this article introduces BeeGOns!, a wireless sensor node following the fog computing paradigm. Our design has been optimized regarding the orchestration of the resources [6]. One of the main challenges of this work is to develop a multisensor node considering the requirements of the sensor metrics and autonomy. This has led us to move from an Arduino system to a higher-performance microprocessor embedded system, in this case, an RPv4. This change in design and development has led to an optimal strategy in which the following contributions can be highlighted.

1) *Smart Design:* The design and development of our solution had as a prerequisite to be able to develop an intelligent system, i.e., autonomy in terms of data orchestration, filtering, and analysis.
2) *Fog Level Integration:* It should be noted that with a microprocessor we can perform first-level data processing tasks. This fact allows us to be able to send alerts generated by complex event processing (CEP) directly from the edge layer as indicated in [6].
3) *Aerodynamic (Packaging) Design:* The packaging design had as a main aim to conveniently place the different components, i.e., avoid any interference between them. In addition, the packaging structure of the node was designed taking into account the data acquisition of air-flow environmental metrics.

Manuscript received 30 August 2022; revised 21 April 2023 and 17 June 2023; accepted 31 July 2023. Date of publication 15 August 2023; date of current version 26 December 2023. This work was supported in part by the Spanish Ministry of Science, Education and Universities; in part by the European Regional Development Fund; in part by the State Research Agency under Grant PID2021-123627OB-C52; and in part by CYTED under Grant 520rt0011. This article was recommended by Associate Editor Q. Zhu. *(Corresponding author: Manuel Castillo-Cara.)*

Michael Vera-Panez and Kewin Cuadros-Claro are with the Information Technologies Department, Universidad Nacional de Ingeniería, Lima 15333, Peru.

Manuel Castillo-Cara is with the Artificial Intelligence Department, Universidad Politécnica de Madrid, 28040 Madrid, Spain (e-mail: manuel.castillo.cara@upm.es).

Luis Orozco-Barbosa is with the Albacete Research Institute of Informatics, Universidad de Castilla-La Mancha (Albacete), 02001 Albacete, Spain.

Digital Object Identifier 10.1109/TCAD.2023.3305575

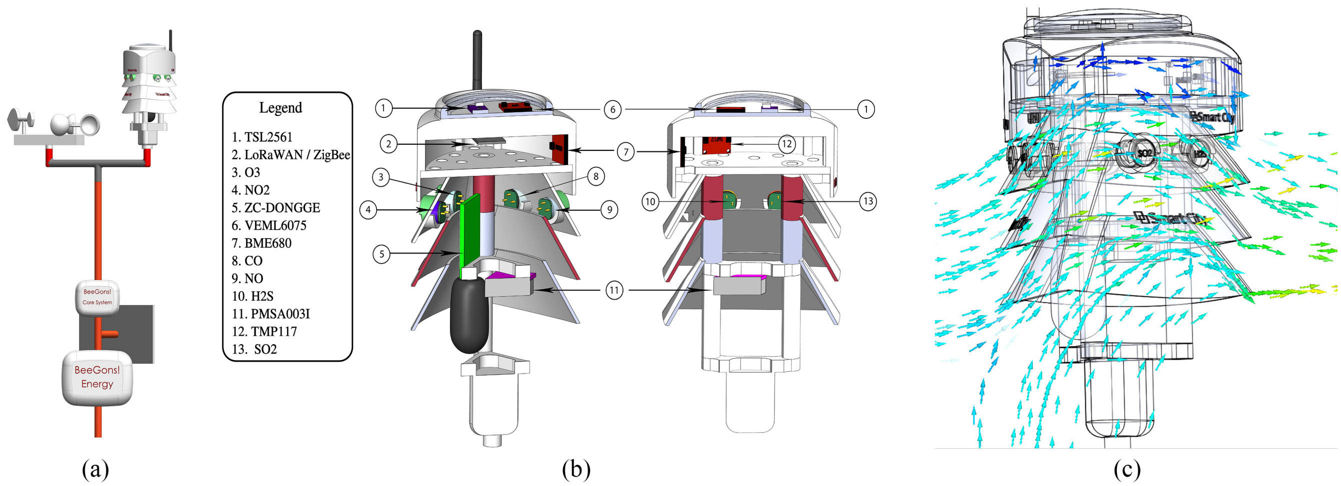

Fig. 1. Design of BeeGOns! with all components and different perspectives of the mechanical design. (a) Complete schematic showing the different components. (b) Cutaway view showing the interior design and sensors layout for air quality module. (c) View of the airflow simulation in the air quality module.

4) *Energy Efficiency:* Both the design of the solar panel for energy acquisition and its integration with the battery have been optimized; making BeeGOns! totally autonomous.
5) *Communications Technology:* We carried out a comparative study of computational and energy consumption of LoRaWAN and ZigBee radio technologies in Section III. Moreover, we also include results for the wired link connecting a BeeGOns! gateway to the cloud.
6) *Open Hardware and Software:* All developed resources are released to the community [15].

In the following, this article is organized as follows. Section I introduces the work and states the main contributions of this work. Section II describes the microcontroller and sensors used to develop BeeGOns. We also describe the design of the mechanical, electronics, and programming components of our solution. In Section III, we mainly examine the computational and energy consumption of the ZigBee and LoRaWAN radio systems and wired link. Section IV finally spells out our conclusions and future research plans.

II. BeeGOns! Design and Development

This section first describes the components that make up BeeGOns!. Then, we describe the different phases of our design methodology.

A. Devices and Protocols

BeeGons! is composed of four main components [see Fig. 1(a)]: 1) an air quality and sound measurement module comprising a total of 13 sensors (top right); 2) a composite weather center (top left); 3) an energy harvesting and storage module (bottom); and 4) a Core System consisting of a main microprocessor (on top of the battery). Table I lists the specifications of the various devices that make up BeeGOns! with RPv4 as the main microprocessor. Alphasense sensors monitor air quality sensitively and with high performance, in contrast to other sensors tested that have the problem of requiring frequent recalibration and deficiencies in the measurement processes. Note that, as for the communications interfaces, we have included two to compare the computational and energy consumption (see Section III).

The interconnection of sensors and RPv4 in BeeGOns! has been carried out using the serial communication protocol I_2C. For the group of analog sensors, the obtained analog signal is converted to a digital signal using an ADC (ADS1115). Hence, the signal obtained at the output of the ADC is passed to an I_2C multiplexer and then connected using the I_2C serial communication protocol to the Core System, which centralizes all the internal and external communications.

TABLE I
Microcontroller and Sensors Implemented in BeeGOns!

Device	Type	Manufacturer	Range	Price (USD)	Consump.
Raspberry Pi 4B	Single-board computer	Sparkfun	—	85	2.5A
ZigBee	Radio frequency	XBee Digi	—	45	210mA
LoRaWAN	Radio frequency	RAK	—	80	15mA
NO_2+ISB	Nitrogen dioxide	Alphasense	[0;16.2]ppm	216	1.5mA
CO+ISB	Carbon monoxide	Alphasense	[0;11,9]ppm	216	1.5mA
SO_2+ISB	Sulfur dioxide	Alphasense	[0;17.2]ppm	216	1.5mA
H_2S+ISB	hydrogen sulphide	Alphasense	[0;3.6]ppm	216	1.5mA
O_3+ISB	Ozone	Alphasense	[0;16.7]ppm	216	1.5mA
NO+ISB	Nitric oxide	Alphasense	[0;9.6]ppm	216	1.5mA
3×Weather meter kit	Anemomenter Wind vane Rain gauge	3×Sparkfun	1.492MPH 0.2794mm	3×80	3×1.5mA
3×BME680	Humidity Presure Temperature	3×Sparkfun	[20;80]%Hr [300;1100]hPa [0-65]°C	3×20.95	3×18mA
ZC-DONGGE	Sound sensor	Taidacent	[30;120]dBA	40	18.9mA
TSL2561	Luminance	ZIO QWIIC	[0.1;40000]Lux	5.9	0.6mA
2×CCS811 TVOC	Organic volatile compound CO_2 air quality	2×Sparkfun	[0;32768]ppm [400;29206]ppm	2×20.95	2×30mA
VEML6075	UVA and UVB	Sparkfun	[0;15]Ind.UV	7.5	480μA
TMP117	Temperature	Sparkfun	[-55;150]°C	13.95	3.5μA
PMSA003I	PM1.0, PM2.5 and PM10	ZIO QWIIC	[0-500]μg/m^3	37.9	100mA
ZHIPS	3D printing material	—	—	80	—
Acrylic	3D printing Acrylic	—	—	15	—

B. Mechanical Design

This phase consisted of the design and construction of the BeeGOns! package. The main task consisted of properly placing the various components on the printed circuit board (PCB) enabling the collection of the various parameters. Hence, the mechanical design was implemented on the basis of the characteristics of each sensor that forms part of the BeeGOns! module. The data acquisition of each sensor is a priority and there must be a minimum error in the reading, i.e., its location in the module has been analyzed and structured empirically, reaching an optimal design. Fig. 1(a) shows the overall structure of BeeGOns! comprised by four main blocks, while Fig. 1(b) shows the placement of the air quality and sound sensors with the position of each sensor (corresponding to the column "device" shown in Table I).

BeeGOns! has been developed empirically after analyzing the measurable behavior of the parameters of each sensor that makes up the module. As shown in Fig. 1(a), the air quality part of the module and the Core System are completely separated. This is a relevant aspect in the design to avoid errors in the acquisition of data from the sensors due to the fact that the RPv4 generates a thermal load that can alter the values acquired by the sensors, e.g., the temperature sensor. Furthermore, the rounded shell was built in transparent acrylic to better protect the sensors from radiation and luminosity, but without obstructing or altering the data acquisition. By distributing each sensor inside the module in a location where the reading of the measurable parameters is efficient.

The aerodynamic design responds to the need for airflow without obstacles. This design will allow the air to enter directly into the interior, where the sensors are protected for their respective reading, which justifies its flake-like structure. Therefore, the air flow hits the inclined part of the disc and enters through the opening at an angle determined by the inclination of the disc, allowing it to pass through the top of the skeleton. The top and discs are shaped following a Reuleaux triangle. This shape allows the airflow to circulate in a circular direction and descend through the other end [see Fig. 1(c)]. In this way, the air is able to pass directly to the air quality measurement sensors, thus achieving an optimum reading for each sensor.

The design was made using SolidWorks software and printed on a 3-D printer. Since BeeGOns! modules will be mainly deployed outdoors, we have designed to support exposure to changing climatic conditions (humidity and high summer temperatures) and to resist impacts of objects and animals to ensure extended durability.

C. Electronic Design

In this phase, we first verified the operation of the various sensors before integrating them into the circuit. BeeGOns! presents a detailed electronic schematic of the electronic PCB implementation integrating all the sensors and the power system, including the battery. One of the main tasks of this phase focused on the design of the Core System, as well as the autonomy of BeeGOns! through the battery and an energy solar harvesting mechanism. As for the Core System, an RPv4 has been used with the following main features: 1) a processing unit and memory enabling the integration of multiple sensors and data orchestration and 2) use of different communication technologies.

The large number of sensors for BeeGOns! as well as RPv4, lead to an energy consumption whose optimization is highly relevant due to the autonomy requirements of BeeGOns!. The energy supply system of BeeGOns! consists of a 20-W solar panel, a 17.5-V open-circuit charger, and a 12-V and 6400-mAh battery. The total consumption of BeeGOns! is 2873 and 2678 mA when using the ZigBee and LoRaWAN interfaces, respectively (see Table I). The system uses a charge/discharge controller to store energy from the solar panel. The initialization, data acquisition, and transmission tasks are scheduled by a Crontab process. Without sufficient power for operation, the charge controller stops and shuts down. Subsequently, the controller is reactivated when the minimum battery voltage is reached. Hence, this starts the process of switching on the Raspberry and automatically executes the main program.

D. Programming Design

This phase consisted of the design and development of the data acquisition algorithm and data transmission. A serial port was enabled to interconnect the onboard sensors with the microcontroller. Therefore, the BeeGOns! programming design was implemented in Python. The developed code performs the reading of each sensor, the interconnection protocols used by the various sensors, and the analog-to-digital conversion of the data sensors as needed. Additionally, a scheduling mechanism was implemented enabling the gathering of the data through the serial port. This latter mechanism creates a list of different sensors whose data have to be collected in a given time and packs them in an array. Finally, a JSON object is created comprising the list of sensor/value pairs. The object is then sent to the core level of the platform.

III. CONSUMPTION-EFFICIENT EVALUATION

In this section, the main outcomes of the computational and energy consumption evaluation analysis are presented considering three different communication architectures.

A. Experimental Scenarios Evaluated

One of the main experimental motivations behind the implementation of the module was to verify the module's autonomy based on the three scenarios depicted in Fig. 2. Therefore, each of the scenarios under evaluation is explained as follows.

1) Scenario 1—Wired Link: In this scenario, no radio data transmission system is used, i.e., there is no WSN. The module is directly connected to the cloud via HTTP. Therefore, the data is collected by the BeeGOns! module and sent to the cloud via a cable connection to the Internet. This baseline scenario allows us to evaluate the performance, mainly the power consumption, required by the wired link.

2) Scenario 2—ZigBee: ZigBee creates a mesh network in which, in this case, there is a Fog and End Node, both are BeeGOns! modules, located at the edge level. That is, the data acquired by BeeGOns! from the end nodes is sent to the coordinator node, which will be in charge of gathering all the packets that arrive and forwarding them as a JSON object to the cloud via HTTP. Note that End Nodes are only collectors, while Fog Nodes are collectors, receivers, and routers [8]. Similarly, the deployment of any service, such as the CEP engine, could be carried out on the Fog Nodes directly, enabling the data analysis of the WSN.

3) Scenario 3—LoRaWAN: LoRaWAN creates a network environment in which the End Nodes are the BeeGOns! modules, which are the information collector nodes. This information is packaged in a JSON object that will be sent to the Gateway. The Gateway collects the JSON object from the End Nodes and sends it to the cloud via HTTP. Note that in this scenario only the End Nodes would be the ones collecting information. Similarly, the deployment of any service, such as the CEP Engine, would have to be carried out at the End Nodes since, as can be seen in Fig. 2, there is no Fog Node to process the WSN information as in a ZigBee network.

B. Testbed Description

For this experiment the air quality module was fully assembled, an ac–dc supply was enlisted to provide sufficient power to operate the

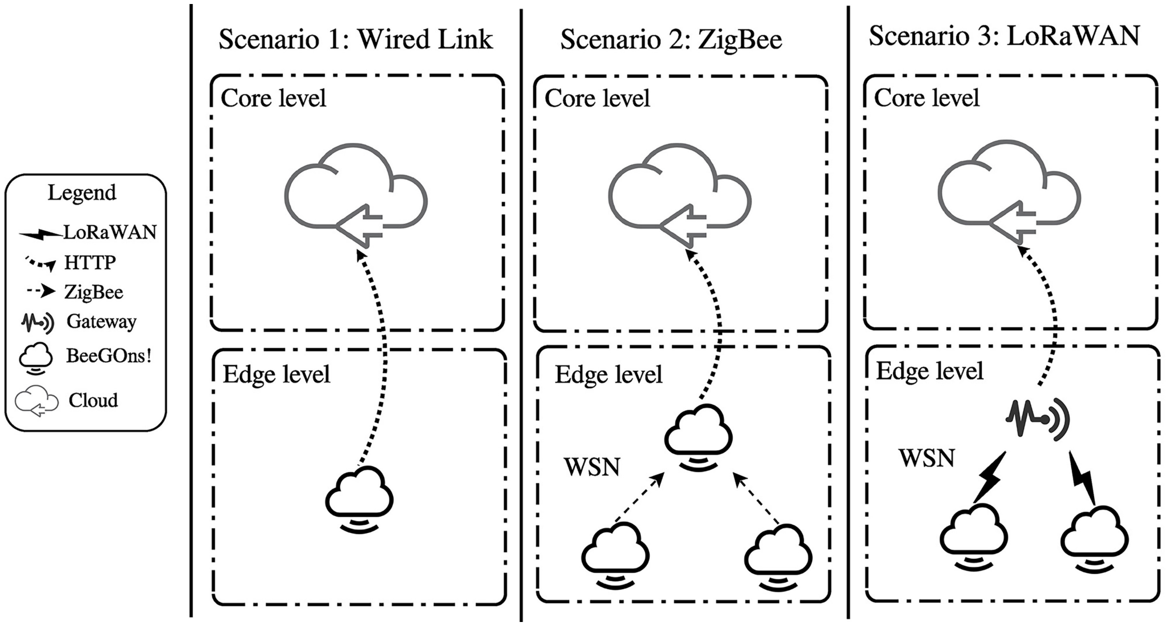

Fig. 2. Overall schema of the experimental scenarios evaluated.

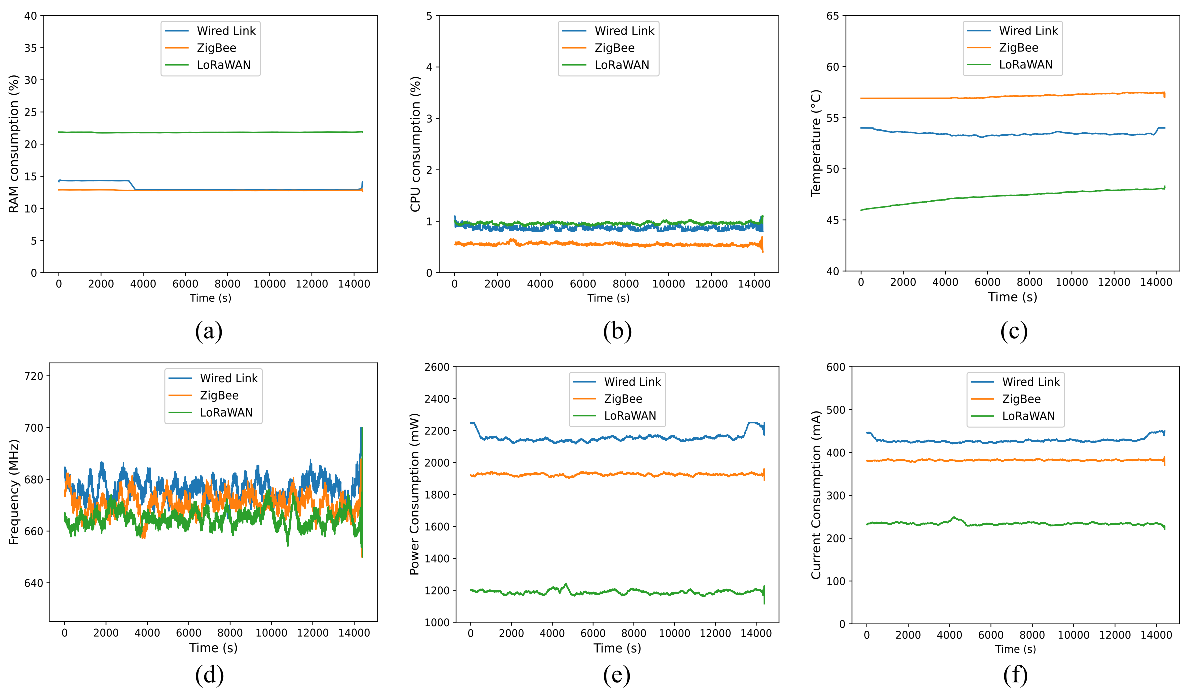

Fig. 3. Performance breakdown of BeeGOns at the edge level. (a) RAM consumption (%). (b) CPU consumption (%). (c) CPU temperature (°C). (d) CPU frequency (MHz). (e) Power consumption (mW). (f) Current consumption (mA).

module with a maximum current of 3 A and an output voltage of 5-V dc. The evaluation was carried out with BeeGOns! as the End Node. Therefore, the computational requirements and energy consumption of the module have been analyzed for the three system configurations.

For energy consumption, the INA 219 sensor was connected in series between the power source and the device to obtain the three metrics of interest: current, voltage, and power consumption of the module. As for the computational, RAM and CPU, utilization rates, we used Sysstat: a Linux open-source software library designed for this purpose.

Finally, all these tasks were programmed using Crontab, with a monitoring interval of 1 s. The transmission rate of the sensor reading was set to one JSON packet per minute. This sampling interval responds to the need to properly detect significant changes in the air quality, as recommended by the sensor manufacturer. The running time of each trial was 4 h. The data reported in the following section corresponds to the average of ten trials.

C. Computational and Energy Consumption

This section reports the computational and energy consumption results of our three testbed configurations (see Fig. 3).

The computational requirements are given in terms of the utilization percentage of the CPU and RAM resources. Regarding RAM space, the wired link and ZigBee communications report similar requirements which correspond to approximately half of the ones needed by LoRaWAN [see Fig. 3(a)]. With respect to CPU utilization, ZigBee has lower requirements. It is worth mentioning that the CPU requirements of all three scenarios are quite low, 1% or less [see Fig. 3(b)]. As seen in Fig. 3(d), the CPU operating frequency is 675MHz. Since our main objective is the design of a Fog computing platform, the computational resources, RAM and CPU, can be mainly allocated to the processing of more intensive computational tasks, such as the storage and data analysis of the collected data. Another result of interest relates to the CPU temperature. As seen in Fig. 3(c), ZigBee reports the highest value 58 °C, a value even surprisingly higher than the one corresponding to the wired link 55 °C; while LoRaWAN reports the best results 45 °C making an excellent candidate as the preferred radio system for such platforms.

Finally, Fig. 3(e) and (f) reports the power (energy per unit of time) and current (minimum current required for the operation of the device) consumption for all three platform setups. The results show that LoRaWAN has much lower power and current consumption requirements than the ZigBee radio system. Note that in the case of the LoRaWAN system, the antenna is connected directly to the GPIO, i.e., it does not use the USB and Ethernet ports. Nevertheless, in the case of the wired link and ZigBee system configurations, these two ports cannot be disabled as they are used to keep the connection and the data transmission processes running.

IV. Conclusion and Open Challenges

In this article, we have developed BeeGOns! a multisensor wireless node. Our design has been fully justified after a review of current trends on the use of IoT technology for a fog computing paradigm in smart city initiatives.

We base our design on the use of open software and open hardware platforms. Therefore, we developed a customized solution using off-the-shelf devices. The design of our solution has addressed the major needs of the various parameters that are being monitored and the operating requirements of the communication, processing, and power supply subsystems.

Our results confirm the superiority of LoRaWAN in terms of the power consumption requirements over ZigBee. We also include an evaluation of the communications services over a wired link. The results show a narrow gap between the requirements of the ZigBee and wired link implementations. These results should be further explored in order to devise energy-saving protocols for battery-operated wireless nodes.

Our immediate research will focus on exploring the use of our solution in the field of Smart City scenarios. The system will support various end-user services based on a CEP system and a flexible LAN/WAN fog computing paradigm. Our immediate research plans include deploying the sensors in a Federated-Fog Computing architecture with the main purpose of fully distributing the data across the fog nodes.

Open Sources

BeeGOns! is an open software/hardware system. All its resources are publicly available at [15].

References

[1] M. Laroui, B. Nour, H. Moungla, M. A. Cherif, H. Afifi, and M. Guizani, “Edge and fog computing for IoT: A survey on current research activities & future directions,” *Comput. Commun.*, vol. 180, no. 1, pp. 210–231, Sep. 2021.

[2] M. A. Ramírez-Moreno et al., “Sensors for sustainable smart cities: A review,” *Appl. Sci.*, vol. 11, no. 17, p. 8198, 2021.

[3] Y. Cui, K. Cao, G. Cao, M. Qiu, and T. Wei, “Client scheduling and resource management for efficient training in heterogeneous IoT-edge federated learning,” *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 41, no. 8, pp. 2407–2420, Aug. 2022.

[4] G. Ortiz et al., “A microservice architecture for real-time IoT data processing: A reusable Web of things approach for smart ports,” *Comput. Stand. Interfaces*, vol. 81, Apr. 2022, Art. no. 103604.

[5] H.-C. Lee and K.-H. Ke, “Monitoring of large-area IoT sensors using a LoRa wireless mesh network system: Design and evaluation,” *IEEE Trans. Instrum. Meas.*, vol. 67, no. 9, pp. 2177–2187, Sep. 2018.

[6] A. Tenorio-Trigoso, M. Castillo-Cara, G. Mondragón-Ruiz, C. Carrión, and B. Caminero, “An analysis of computational resources of event-driven streaming data flow for Internet of Things: A case study,” *Comput. J.*, vol. 66, no. 1, pp. 47–60, Oct. 2021.

[7] M. El-Hosseini, H. ZainEldin, H. Arafat, and M. Badawy, “A fire detection model based on power-aware scheduling for IoT-sensors in smart cities with partial coverage,” *J. Ambient Intell. Humanized Comput.*, vol. 12, no. 2, pp. 2629–2648, 2021.

[8] M. Castillo-Cara, E. Huaranga-Junco, M. Quispe-Montesinos, L. Orozco-Barbosa, and E. A. Antúnez, “FROG: A robust and green wireless sensor node for fog computing platforms,” *J. Sens.*, vol. 2018, Ar. 2018, Art. no. 3406858.

[9] F. Akhter, S. Khadivizand, H. R. Siddiquei, M. E. E. Alahi, and S. Mukhopadhyay, “IoT enabled intelligent sensor node for smart city: Pedestrian counting and ambient monitoring,” *Sensors*, vol. 19, no. 15, p. 3374, 2019.

[10] C. Correa, D. Dujovne, and F. Bolaño, “Design and implementation of an embedded edge-processing water quality monitoring system for underground waters,” *IEEE Embedded Syst. Lett.*, vol. 15, no. 2, pp. 81–84, Jun. 2023.

[11] A. Palla-Papavlu, S. I. Voicu, and M. Dinescu, “Sensitive materials and coating technologies for surface acoustic wave sensors,” *Chemosensors*, vol. 9, no. 5, p. 105, 2021.

[12] D. Ayala-Ruiz, A. C. Atoche, E. Ruiz-Ibarra, E. O. de la Rosa, and J. Vázquez Castillo, “A self-powered PMFC-based wireless sensor node for smart city applications,” *Wireless Commun. Mobile Comput.*, vol. 2019, Jun. 2019, Art. no. 8986302.

[13] C. Catlett et al., “Hands-on computer science: The array of things experimental urban instrument,” *Comput. Sci. Eng.*, vol. 24, no. 1, pp. 57–63, 2022.

[14] S. Metia, H. A. D. Nguyen, and Q. P. Ha, “IoT-enabled wireless sensor networks for air pollution monitoring with extended fractional-order Kalman filtering,” *Sensors*, vol. 21, no. 16, p. 5313, 2021.

[15] M. Castillo-Cara and M. Vera-Panez. “BeeGOns! Wireless sensor node.” Mar. 2023. [Online]. Available: https://doi.org/10.5281/zenodo.6343059