

Power Consumption of Prototyping Boards for Smart Room Temperature Monitoring

Petros Spachos*, Liang Song[§], and Stefano Gregori*

*School of Engineering, University of Guelph, Canada

[§]School of Electronic and Information Engineering, Henan University of Science and Technology, People's Republic of China

Abstract—The introduction of Internet of Things (IoT) and smart cities is widely viewed as a great opportunity that stimulated the modern view of wireless sensor networks (WSNs). Up to this point, WSNs were application-driven systems with well defined challenges and requirements. The idea of IoT and smart cities increased the amount of available data and magnified the hardware and software challenges of sensor integration. Every day the number of the sensors that connect to the internet increases, providing still unexplored capabilities, but also posing significant challenges. One of them, which is common to traditional WSNs, is still the power consumption. Especially for monitoring inaccessible locations, the lifetime of a node and of the whole network is crucial. At the same time, in large-scale networks not only the power consumption, but also the cost of each node is important. Prototyping boards such as Arduino, RedBoard, and Raspberry Pi can alleviate the problem by providing low-cost and low-power nodes. This paper evaluates the power consumption of the above three prototyping boards in a WSN application. Experiments were conducted and the power consumption of each board was measured under three scenarios. The results are a useful indicator for the expected power consumption of smart-city application deployments.

I. INTRODUCTION

Day after day more and more electronic devices are connected to the Internet. Since most of these devices have at least one sensing element, the interconnections of computer systems to the physical world grow incessantly. This concept underpins the Internet of Things (IoT), a development of the Internet in which common objects send and receive data through the network. In the near future, many wireless sensor networks (WSNs) will be integrated into the IoT and consequently into smart cities.

The future Internet, designed as an IoT, will be a worldwide network of uniquely-addressable objects via standard communication protocols. Sensor nodes will join the Internet dynamically and use it collaboratively to accomplish their tasks. Therefore the sensing infrastructure will have a key function in the IoT and great research opportunities.

Sensors will also have a crucial role in smart cities. Green cities, which include aspects of sustainable transport, citizen sensing, urban agriculture, energy and water management, and other smart city projects, are emerging in the attempt to improve sustainability and quality of services through greater connectivity.

At the same time, the sensor node market continues to grow rapidly, while the cost to build custom nodes decreases due to technology scaling. Advances in integration and manufacturing technology have opened the door for increasingly sophisticated sensors to be used in environmental monitoring research. Moreover, the use of embedded sensors in smartphones, phablets, tablets and other mobile devices is enabling a number of popular applications.

The flexibility of sensor nodes permits fast time-to-market, which is crucial in the development of new WSN applications. From the cost perspective, embedded sensors in mobile devices are already superior to application specific hardware. However, the programmability of sensors and their integration into multiple applications involve an overhead. More research is needed to narrow the gap between sensors and their potential applications in order to broaden the scope of the applications and foster more innovation. Enabling greater innovation will ultimately benefit the economy and society at large.

Beyond improving sensor integration, WSNs in IoT and smart cities enable yet unexplored applications that cannot be conceived using specific or dedicated hardware, especially in the context of mobile devices. A flexible embedded sensor framework can be programmed and reprogrammed in real-time, based on application needs, a unique characteristic that can be exploited for a variety of purposes, including improving application speed, power and cost.

In this paper, three popular prototyping boards are examined: Arduino, RedBoard, and Raspberry Pi. The boards have similar cost, however there are quite a few differences in their hardware and capabilities. During the experimentation the boards perform exactly the same operation. We examined specifically the power consumption under a simple smart-city application, which is smart room temperature monitoring. We present experimental results of the energy requirements of the boards under three different scenarios: the boards in idle mode, acquiring the sensor data, and transmitting the data. The results can be used as an indicator for future smart-city application development based on the expected power requirements.

The main contributions of this paper are listed below:

- We provide extensive experimentation and analytic results on the energy requirements for three popular components for a temperature monitoring application.

- We propose to use a specific configuration to minimize the energy requirements in the examined scenario.
- Through comprehensive performance evaluation, we demonstrate the efficiency of each board in a smart room temperature monitoring scenario without further data processing and hardware modifications. The evaluation results can be used for application in smart homes and smart buildings.

The rest of this paper is organized as follows: The related work is reviewed in Section II. In Section III, the experimental method is described including the experimental setup and procedure. The performance evaluation and analysis is presented in Section IV with the experimental results and a brief discussion. In Section V is the conclusion of this work.

II. RELATED WORK

The study of urban sensing, IoT, and smart cities still remains a niche. The transformation to smarter cities requires innovation in management, planning, and operations [1]. Changing the status quo always requires significant efforts and several ongoing projects worldwide illustrate the challenges of this transformation, as well as the opportunities and advantages [2]. The initial endeavours of the research community towards the new challenges lead to a variety of general architectures and protocols, mainly from the area of WSNs [3], [4]. Another popular research initiative is in the exploration of the special features and problems of the integration of IoT and smart cities [5]–[7]. At the same time, the businesses centred around smart cities are slow in taking off, mainly because of the lack of a general sustainable framework [8].

Managing energy systems to improve efficiency is an example of the challenges for smart cities in which the integration through WSNs of different subsystems, currently independent, can make the difference in terms of energy savings [9]. Starting from reducing the energy footprint of large buildings, the approach can be scaled towards a smart city with substantial benefits [10].

The power consumption monitoring of wireless deployments is also an active research area. As an example, a power measurement and control board named Energino has been used to monitor the power consumption of the WiFi adapter of a PCEngines ALIX 3D2 (500 MHz x86 CPU, 256 MB of RAM) and a model for WiFi traffic was generated [11]. The measurements taken using Energino had a power resolution of 135 mW and time resolution of 10 ms, and included also the power consumption of the ZigBee communication chips. The measurements were focused on deriving the model for the generated traffic, rather than analyzing of the computational complexity of WiFi encryption and the Ethernet interface.

Different techniques have been proposed to minimize the power consumption of XBee, which is a popular solution for prototyping. A data fusion mechanism, called local data differentiation, was introduced to reduce energy use [12].

In our previous work, a prototype was developed for smart room monitoring [13]–[15]. In this work, we focus on the



(a) Arduino Uno [17]. (b) RedBoard [18]. (c) Raspberry Pi 2 [19].

Fig. 1. Prototyping boards' photos.

TABLE I
PROTOTYPING BOARDS' SPECIFICATIONS.

Board	Arduino Uno [17]	RedBoard [18]	Raspberry Pi 2 [19]
CPU	ATmega328 (8-bit) with Optiboot	ATmega328 (8-bit) with Optiboot	ARM Cortex-A53 (64-bit, quad-core)
Clock speed	16 MHz	16 MHz	900 MHz
Memory	2 kB RAM, 32 kB Flash, 1 kB EEPROM	2 kB RAM, 32 kB Flash, 1 kB EEPROM	1 GB RAM
GPU	none	none	VideoCore IV
External storage	MicroSD	MicroSD	MicroSD
Networking	through shields	through shields	10/100 Mb/s ethernet
Power source	6–20 V (2-mm jack) 5 V (USB B)	6–20 V (2-mm jack) 5 V (USB mini-B)	5 V (USB micro-B)
Dimensions	69 mm × 53 mm	69 mm × 53 mm	86 mm × 57 mm
Weight	25 g	23 g	45 g
Appr. price (USD)	\$30	\$25	\$35

power usage of three prototyping boards for a simple WSN for smart home temperature monitoring. We explore the power consumption of the boards with no extra configuration which might minimize the energy requirements. This work can be used as an upper limit indicator of the energy consumption of the boards without further data processing and hardware modifications.

III. EXPERIMENTAL METHOD

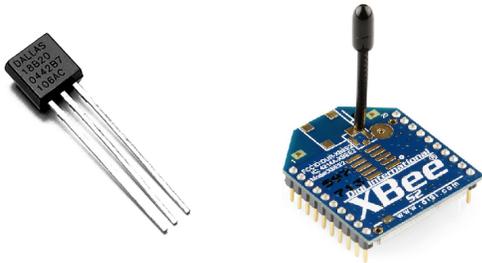
In this section, we describe the experimental setup and the hardware tools we used, the scenarios that were examined followed by the experimental procedure for the power measurement.

A. Experimental Setup

Each experiment was conducted using the Power Monitor from Monsoon Solutions [16] controlled by a laptop computer running Windows 7 with an Intel Core i7 processor and 4 GB RAM.

Three prototyping boards were characterized: Arduino Uno board [17], SparkFun RedBoard [18], and Raspberry Pi 2 Model B [19], shown in Fig. 1. All the three are popular prototyping boards, due to their easy to use features, low cost and small size. The specifications of each board are summarized in Table I.

It is clear that the boards have different hardware configuration and capabilities. Based on the application, the user might select one board over the other. In this work, we examine an



(a) DS18B20 [20]. (b) XBee [21].

Fig. 2. Temperature sensor and ZigBee module with wire antenna.

application that all the three boards can perform. Our goal is to examine the power usage of the boards under a simple and popular application for smart buildings.

For the experiments, we also used the Maxim Integrated DS18B20 digital thermometer [20] shown in Fig. 2a as temperature sensor, and the Digi XBee 2-mW module with wire antenna (ZigBee Mesh) [21] shown in Fig. 2b for the wireless communication.

B. Scenarios

We measured the instantaneous power consumption of the boards under three different scenarios:

- **Scenario 1: Board in idle mode.** The board is powered up through the power monitor. After the necessary initialization time, the board goes in idle mode. There are no processes running on the board while there are no other peripherals connected to it.
- **Scenario 2: Board connected with the sensor acquiring data.** The sensor is wired to the microcontroller. After the necessary initialization phase and the sensor warm up time, a script is running and reads the data from the sensor. There are no other processes running nor peripherals connected to the board.
- **Scenario 3: Board transmitting the data over the WiFi shield.** The sensor and the antenna are wired to the board. Then, a network consisting of 5 relay nodes is deployed. The relay nodes have the same board and antenna as the source node and forward all the data to a control room. The control room collects and stores all the data. During initialization the source sets up the path towards the control room. At the source board, a script is running which reads the sensor data and transmits them wirelessly through the antenna. There is no other process running or peripheral connected to the board. For this scenario, the source node was deployed in an office and the control room was another office at the same floor in the Richards Building on the University of Guelph campus. The layout can be seen in Fig. 3.

The PowerTool software and Power Monitor hardware can analyze the power on any device that uses up to 4.55 V battery. The power monitor can measure data on three channels: Main, USB and Auxiliary. For this experiment, we have used the Main channel only.

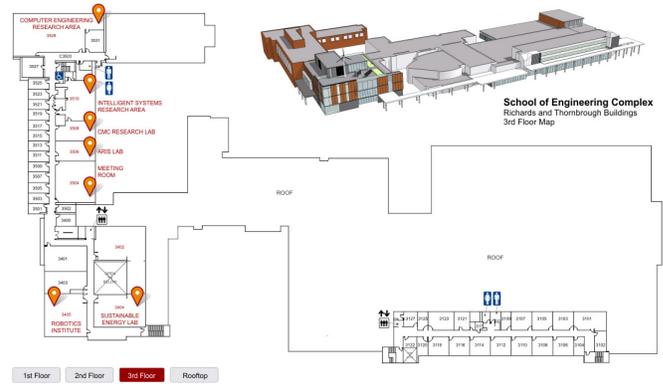


Fig. 3. Scenario 3 network topology [22].

C. Experimental Procedure

In total, we conducted nine measurements, three for every board, following the different scenarios described in the previous section.

At the beginning of each measurement there was an initialization phase for 60 s. This is the necessary time for the board to go to idle mode (Scenario 1), the sensor to warm up (Scenario 2) and to establish connection with a neighbour node (Scenario 3). After the initialization phase, we measured the current, the voltage and the power consumption every 2 ms. Each board was tested for 60 s.

IV. PERFORMANCE EVALUATION AND ANALYSIS

In this section we present the experimental results followed by a brief discussion.

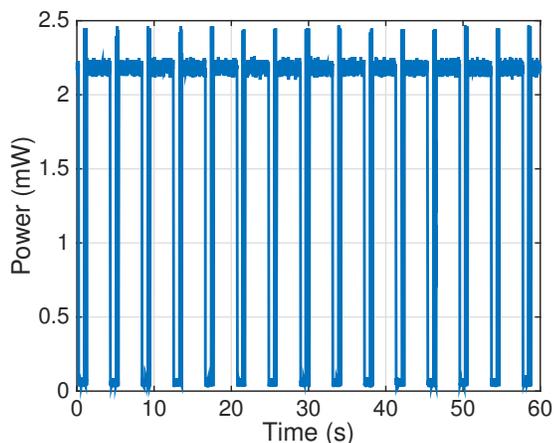
A. Experimental Results

Initially we measured the power consumption of the temperature sensor and the ZigBee module alone.

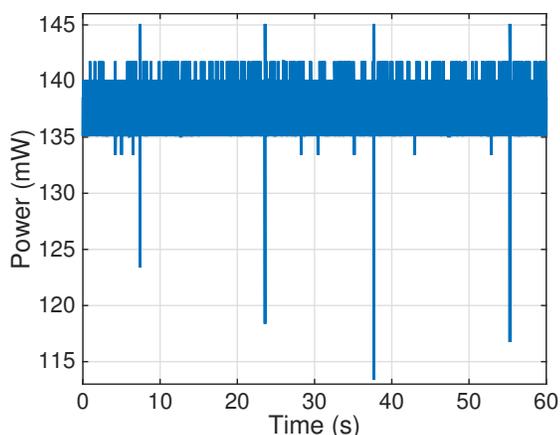
The power consumption of the sensor is shown in Fig. 4a. As it is expected, the power consumption increases during the data acquisition while it drops to zero when the sensor is not operating. On average, the power consumption of the sensor is 2.19 mW. The measurements are taken after the necessary warm up stage of the sensor.

The power consumption of the ZigBee module during transmission is shown in Fig. 4b. The measurements took place after establishing a connection with a neighbour node. On average, the power consumption is 139.1 mW.

Next, we measured the voltage, current, and power for the different scenarios. Table IIa shows the average measurements while Fig. 5 shows the measurements for the 60 s period. RedBoard consumes less power over all while Raspberry Pi 2 consumes the most power. The large difference is probably due to the clock speed and the number of supporting chips on each board. The Raspberry Pi 2 has additional peripheral chips, including an analog-to-digital converter, which all consume power. Even when the board is in idle mode, the power consumption is higher than the other two boards.



(a) Power consumption of the DS18B20 sensor.



(b) Power consumption of the XBee module.

Fig. 4. Power consumption of the temperature sensor and the ZigBee module.

As we connect the sensor, the increase in power consumption is almost negligible. Table IIb shows the average measurements while Fig. 6 shows the measurements for the 60 s period. RedBoard consumes less power over all while Raspberry Pi 2 consumes the most power. This is expected since the port the sensor is connected is already powered from the board even in idle mode and the power requirement of the sensor is negligible in comparison with the requirements of the board. In addition, the script that is running just reads the data, hence, it does not affect the power requirements.

When data transmission is started, the power consumption increases. Table IIc shows the average measurements, while Fig. 7 shows the measurements for the 60 s period. Again, RedBoard consumes less power over all while Raspberry Pi 2 consumes the most power. The results show that, when the boards read the sensor data and transmit them over WiFi, the power consumption is significantly higher.

Raspberry Pi 2 keeps consuming the most power overall and the RedBoard has the lowest power consumption. Arduino Uno total power consumption increases only according to the

TABLE II
AVERAGE VOLTAGE, CURRENT, AND POWER.

Board	Arduino Uno	RedBoard	Raspberry Pi 2
Voltage (V)	3.34	3.34	3.33
Current (mA)	25.17	11.72	295.6
Power (mW)	83.93	39.09	985.6

(a) Scenario 1.

Board	Arduino Uno	RedBoard	Raspberry Pi 2
Voltage (V)	3.34	3.34	3.33
Current (mA)	25.42	11.94	295.7
Power (mW)	84.80	39.83	985.8

(b) Scenario 2.

Board	Arduino Uno	RedBoard	Raspberry Pi 2
Voltage (V)	3.34	3.34	3.33
Current (mA)	66.90	54.05	339.5
Power (mW)	223.2	180.3	1133

(c) Scenario 3.

Board	Arduino Uno	RedBoard	Raspberry Pi 2
$\bar{P}_2 - \bar{P}_1$ (mW)	0.87	0.74	0.17
$\bar{P}_3 - \bar{P}_2$ (mW)	138.4	140.5	146.9

(d) Average power increase from scenario 1 to 2 ($\bar{P}_2 - \bar{P}_1$), and from scenario 2 to 3 ($\bar{P}_3 - \bar{P}_2$).

power requirements of the ZigBee module and the sensor. The power of the other two boards increases a bit more. This indicates that the other boards need to power up more chips in order to set up the network and transmit the data.

B. Discussion

According to the results, RedBoard has the lowest power consumption over the other two boards for a simple application such as smart room temperature monitoring. Arduino has a bit higher power requirements, while Raspberry Pi 2 has the highest power requirements overall. This is mainly because of the different clock speed and additional chips on each board. The experiments were conducted with no other processes running on the boards nor any peripheral devices connected to the boards. There are techniques such as data fusion which can minimize the power usage of the boards, but they require further processing.

The average consumption of every board under the difference scenarios is shown in Fig. 8. It is interesting to notice the big increase in the power consumption for all the three boards, as they start transmitting the data. Also, the difference between Raspberry Pi 2 and the other two boards is high in all the three examined scenarios. Another important aspect is that the difference between Scenario 1 and Scenario 2 is almost negligible.

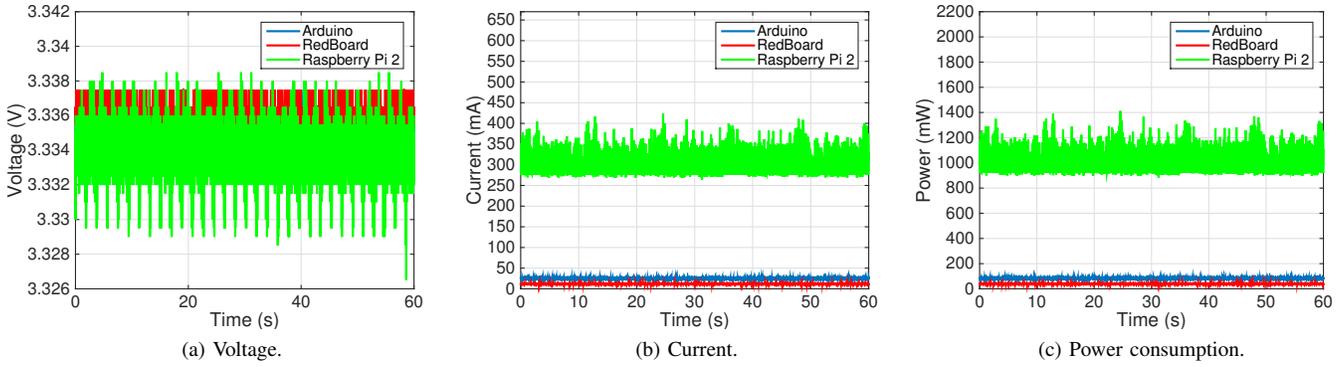


Fig. 5. Scenario 1: Board in idle mode.

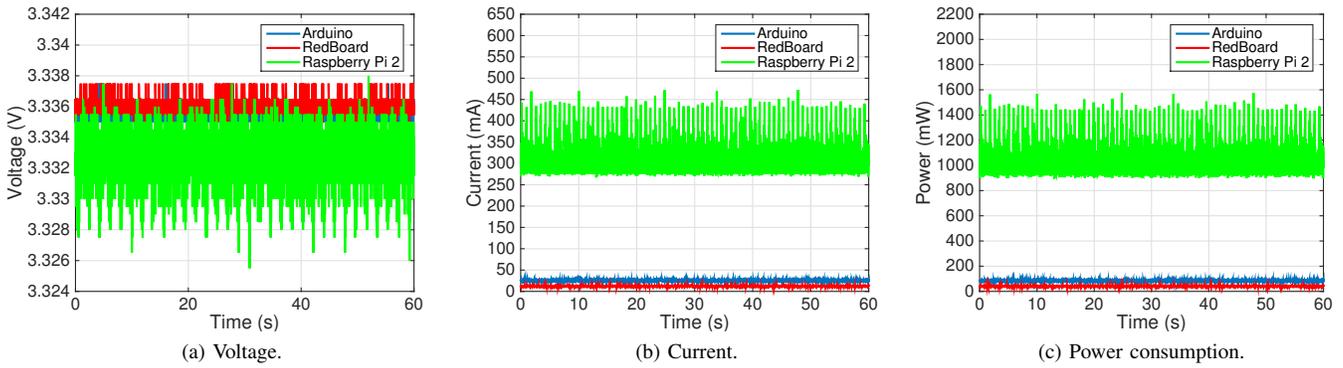


Fig. 6. Scenario 2: Board acquiring data from the sensor.

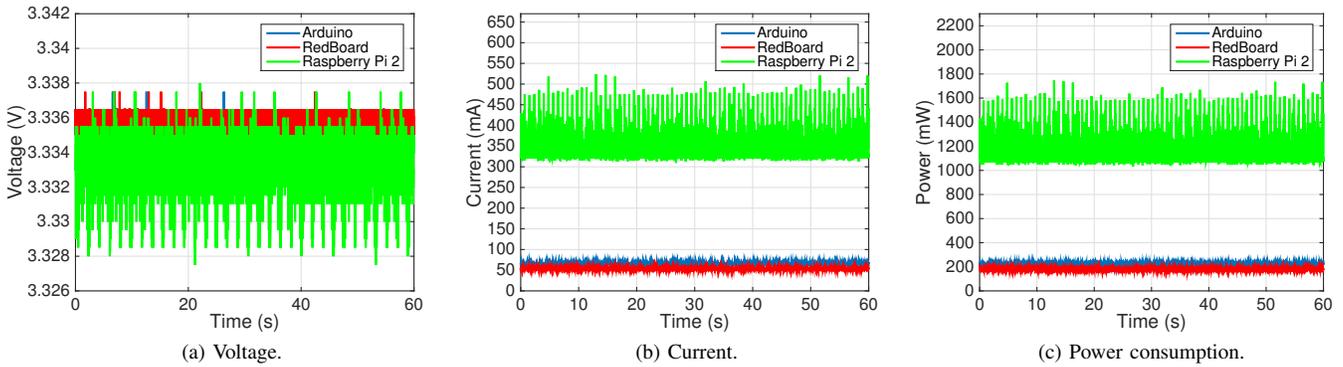


Fig. 7. Scenario 3: Board transmitting the data over the WiFi shield.

Table II summarizes the increase in the power consumption of each board as we move from Scenario 1 to 2 and from Scenario 2 to 3. The detailed results in this table verify the negligible difference in the power consumption between Scenario 1 and 2. Also, according to the experimental results, when sensor data is acquired, the smallest increase in the power consumption comes from the Raspberry Pi 2. However, Raspberry Pi 2 already has high power consumption, hence although the increase is small when sensor data is acquired, it still consumes too much power in comparison with the other

boards.

When the data is transmitted, the smallest increase comes from the Arduino while Raspberry Pi 2 has the highest increase. RedBoard has the second smallest increase and along with the small amount of required energy in the previous mode, make this board the most energy efficient. These experimental results can provide useful insights for future experiments with these boards.

Moreover, the results can be used as an indicator of the power requirements for similar smart cities application. It is

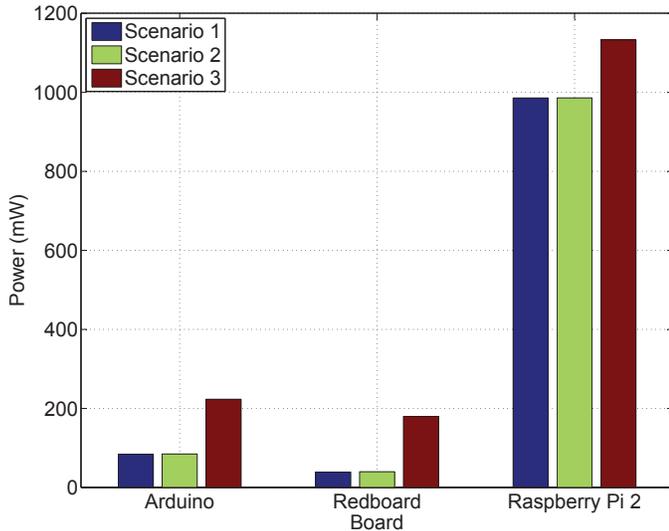


Fig. 8. Power consumption of the boards under different scenarios.

important to notice that the nodes, although having similar costs, have different capabilities, and therefore they are suitable for different smart-city applications.

V. CONCLUSION

In this work, we focused on the energy requirements of three different prototyping boards to set up a simple WSN for temperature monitoring. The boards that were tested have similar costs but different capabilities. The boards were examined under three different scenarios: in idle mode, when they are acquiring sensor data, and when they are transmitting the data. Based on the experimental results, for a simple WSN such as the one examined, RedBoard has the lowest power consumption over the other two boards for the three experimental scenarios. Insights regarding the power increase between the different scenarios are also interesting. The power consumption along with the application requirements and the deployment cost should be considered for efficient WSN deployment in smart city applications.

Further parameters should be examined in the future, including the external temperature of each board as well as any memory requirements.

ACKNOWLEDGMENT

This work was partially supported by the Natural Sciences and Engineering Research Council of Canada through grant EGP 493708-16 and by OMesh Networks Inc.

REFERENCES

- [1] J. Lin, H. Bannazadeh, P. Spachos, and A. Leon-Garcia, *SAVI vCPE and Internet of Things*. Cham: Springer, 2015, vol. 159, pp. 18–25. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-27072-2_3
- [2] M. Naphade, G. Banavar, C. Harrison, J. Paraszczak, and R. Morris, “Smarter cities and their innovation challenges,” *Computer*, vol. 44, no. 6, pp. 32–39, Jun. 2011.
- [3] P. Bellavista, G. Cardone, A. Corradi, and L. Foschini, “Convergence of MANET and WSN in IoT urban scenarios,” *IEEE Sensors J.*, vol. 13, no. 10, pp. 3558–3567, Oct. 2013.
- [4] F.-J. Wu and H. Lim, “Urbanmobilitysense: A user-centric participatory sensing system for transportation activity surveys,” *IEEE Sensors J.*, vol. 14, no. 12, pp. 4165–4174, Dec 2014.
- [5] C. Mulligan and M. Olsson, “Architectural implications of smart city business models: an evolutionary perspective,” *IEEE Commun. Mag.*, vol. 51, no. 6, pp. 80–85, Jun. 2013.
- [6] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, “Internet of things for smart cities,” *IEEE Internet Things J.*, vol. 1, no. 1, pp. 22–32, Feb. 2014.
- [7] J. Jin, J. Gubbi, S. Marusic, and M. Palaniswami, “An information framework for creating a smart city through internet of things,” *IEEE Internet Things J.*, vol. 1, no. 2, pp. 112–121, Apr. 2014.
- [8] I. Vilajosana, J. Llosa, B. Martinez, M. Domingo-Prieto, A. Angles, and X. Vilajosana, “Bootstrapping smart cities through a self-sustainable model based on big data flows,” *IEEE Commun. Mag.*, vol. 51, no. 6, pp. 128–134, Jun. 2013.
- [9] M. Brenna, M. Falvo, F. Foidelli, L. Martirano, F. Massaro, D. Poli, and A. Vaccaro, “Challenges in energy systems for the smart-cities of the future,” in *Proc. IEEE Int. Energy Conf. Exhibition (ENERGYCON)*, 2012, pp. 755–762.
- [10] T. Weng and Y. Agarwal, “From buildings to smart buildings—sensing and actuation to improve energy efficiency,” *IEEE Des. Test Comput.*, vol. 29, no. 4, pp. 36–44, Aug. 2012.
- [11] K. Gomez, R. Riggio, T. Rasheed, D. Miorandi, and F. Granelli, “Energino: A hardware and software solution for energy consumption monitoring,” in *Proc. Int. Symp. Modeling Optimization Mobile, Ad Hoc Wireless Netw. (WiOpt)*, 2012, pp. 311–317.
- [12] Y. F. Solahuddin and W. Ismail, “Data fusion for reducing power consumption in Arduino-Xbee wireless sensor network platform,” in *Proc. Int. Conf. Comput. Inf. Sci. (ICCOINS)*, 2014, pp. 1–6.
- [13] P. Spachos and D. Hatzinakos, “Real-time indoor carbon dioxide monitoring through cognitive wireless sensor networks,” *IEEE Sensors J.*, vol. 16, no. 2, pp. 506–514, Jan. 2016.
- [14] A. Shaltout, P. Spachos, and S. Gregori, “Power management modelling of a photovoltaic system for a wireless sensor network,” in *Proc. IEEE Int. Workshop Comput. Aided Modelling Des. Commun. Links Netw. (CAMAD)*, 2016, pp. 201–206.
- [15] P. Spachos, J. Lin, H. Bannazadeh, and A. Leon-Garcia, “Smart room monitoring through wireless sensor networks in software defined infrastructures,” in *Proc. IEEE Int. Conf. Cloud Networking (CloudNet)*, 2015, pp. 216–218.
- [16] Monsoon Power Monitor. (retrieved: 2017-04-30). [Online]. Available: <http://www.monsoon.com/LabEquipment/PowerMonitor/>
- [17] Arduino UNO. (retrieved: 2017-04-30). [Online]. Available: <https://www.arduino.cc/en/Main/ArduinoBoardUno>
- [18] SparkFun RedBoard. (retrieved: 2017-04-30). [Online]. Available: <https://www.sparkfun.com/products/13975>
- [19] Raspberry Pi 2 Model B. (retrieved: 2017-04-30). [Online]. Available: <https://www.raspberrypi.org/products/raspberrypi-2-model-b/>
- [20] Maxim Integrated DS18B20. (retrieved: 2017-04-30). [Online]. Available: <https://www.maximintegrated.com/en/products/analog/sensors-and-sensor-interface/DS18B20.html>
- [21] Digi XBee Series 2. (retrieved: 2017-04-30). [Online]. Available: <http://www.digi.com/products/xbee-rf-solutions/rf-modules/xbee-zigbee>
- [22] Richards Building floor plan, University of Guelph. (retrieved: 2017-04-30). [Online]. Available: <https://www.uoguelph.ca/engineering/>