ECAI 2018 - International Conference – 10th Edition
Electronics, Computers and Artificial Intelligence
28 June -30 June, 2018, Iasi, ROMÂNIA

Data Compression in Wireless Sensor Nodes with LoRa

D. I. Săcăleanu, R. Popescu, I. P. Manciu, and L. A. Perişoară
Faculty of Electronics, Telecommunication and Information Technology
University Politehnica of Bucharest, Romania
dragos.sacaleanu@ceospacetech.pub.ro, razvan.popescu23@yahoo.ro,
ipmanciu@yahoo.co.uk, lucian.perisoara@upb.ro

Abstract – **In Wireless Sensor Networks (WSN), LoRa technology permits the data communication on large areas, allowing to eliminate multi-hop data transmission that requires a very precise synchronization. Due to these advantages it is being used more and more in practical solution through LoRaWAN protocol. However, although it is a low power protocol, comparing with other data transmission protocols used in WSN, the timeframe for transmitting data is significantly longer and the energy consumption is increased. Thereby, reducing the data volume through data compression has a major impact on node current consumption. This paper presents results of data compression in wireless sensor nodes that uses LoRa technology to transmit data. An energy consume comparison is made with other data communication protocols used in WSN, like ZigBee or Enhanced ShockBurst.**

Keywords – wireless sensor networks, LoRa, data compression, energy saving

I. INTRODUCTION

THE increasing number of interconnected devices in the expanding world of communications has influenced the concept of machines-to-machines communication (M2M) by producing a new generation of sensors and technologies. Considering this, WSN generates an increased interest from industrial and research perspectives [1].

WSN can accommodate a huge number of inexpensive, low-powered sensor nodes disseminated over a certain area. Due to the limited energy source, one of the main research directions in this domain is to reduce the current consumption. Many wireless technologies and protocols were developed in order to respond to an application market with special request on mobility and long-term autonomy. The efforts made to reduce energy consumption are essential to optimize WSN function. Energy efficiency translates into a higher lifetime of the sensor nodes and of the network as a whole.

In WSN, conventional short distance protocols used in wireless communications, like Wi-Fi or Bluetooth, can be applied [2]. A specially developed wireless protocol for WSN is ZigBee. This is a standard-based network protocol supported by the ZigBee Alliance that uses the transport services of the IEEE 802.15.4 network specification where adds specific functionalities [2], [3]. The protocol stack used in wireless sensors consists of physical layer, data link layer, network layer and application layer. ZigBee is used for low power, low data rate, and close proximity wireless networks. Other protocols were designed to be implemented on dedicated transceivers. The Enhanced ShockBurst is a lightweight, proprietary protocol stack developed by Nordic Semiconductor that is used on ultra-low power performance wireless system-on-a-chip (SoC) and connectivity devices for the 2.4 GHz ISM band [4].

Recently, the attention was focused on developing low-power, long range communication protocols [5] where are uses technologies like LoRa or SigFox [6]. LoRa is a technology developed by Semtech with spread spectrum modulation technique. It is a derivative of Chirp Spread Spectrum (CSS) with integrated Forward Error Correction (FEC). The technology offers a very compelling mix of long range, low power consumption and secure data transmission in the unlicensed radio spectrum of ISM bands. The protocol specifications built by LoRa Alliance [7] on top of LoRa technology is LoRaWAN, and enables low power, wide area communication between remote sensors and gateways connected in WSN [8].

The highest amount of energy spent by a wireless sensor node is for data communication [9]. Besides low power communication protocols, the energy efficiency can be improved by reducing the data volume, allowing the transceiver to enter in sleep mode as soon as possible. The most common solutions implemented in WSN are data compression or/and data aggregation [10].

This paper presents the impact of data compression on the wireless sensor nodes lifetime for LoRa technology, comparing with ZigBee and Enhanced ShockBurst. The current consumption was measured during data transmission both with data compression and without data compression. Section II presents the hardware platforms used for the experiments. The framework for data compression is described in Section III. In Section IV, the obtained results are presented, while the Section V concludes the paper.

978-1-5386-4901-5/18/$31.00 ©2018 IEEE

II. Hardware Platforms Used for Testing

To study the impact of data compression in WSN, three different wireless sensor nodes were tested, based on three different data communication protocols: ZigBee, Enhanced ShockBurst and LoRa.

A. DASMote node with ZigBee

To test the energy saving algorithm with the ZigBee communication protocol, we used our own developed node, called DASMote (node for Data Acquisition Systems), see Fig. 1. It was created considering the energy constraints of a wireless sensor node and the applicability for different practical monitoring solutions [11].

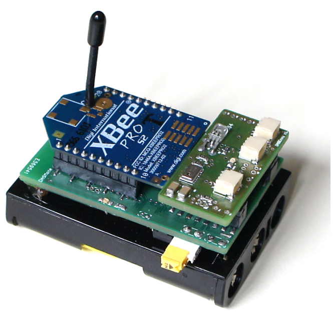

Figure 1. The DASMote node with the XBee transceiver.

To optimize the current consumption of the node, low power components were integrated. The main electronic component is the MSP430F2618 microcontroller from Texas Instruments. The implementation of the ZigBee protocol was achieved using the XBee transceivers family from Digi, which includes models with data communication range from 50 m (XBee ZB) up to 1500 m line-of-sight (XBee Pro S2). Depending on the type of application, a proper model of the transceiver can be attached in the sockets.

The DASMote node can be adapted to various applications using different data acquisition interfaces that can be connected through two 16 contacts sockets. For the performed tests we used an environmental parameters interface, called DASEnvy. The acquired parameters and the sensors are summarized in Table I. For each parameter, the data is acquired on 2 bytes. The attached transceiver is a XBee Pro S2 with integrated wire antenna, settled up at the maximum power level for long range transmission. The power was provided by 3 x 1.5 V AAA batteries.

TABLE I. Used sensors for the DASMote node

Measured Parameter	**Sensor**
Air Temperature	SENSIRION SHT75
Air Humidity	
Soil Temperature	DALLAS DS18B20
Soil Humidity	DECAGON EC-5
Barometric Pressure	FREESCALE MPL115A1
Light Intensity	SHARP GA1A2S100LY
Node Temperature	TexasInstruments TMP102

B. Arduino node with Enhanced ShockBurst

To test the energy saving algorithm with the Enhanced ShockBurst (ESB) protocol, we also developed a low cost, low power node that uses off-the-shelf modules: Arduino Pro Mini 3.3 V development board, nRF24L01 transceiver and own connecting interface, see Fig. 2 [12].

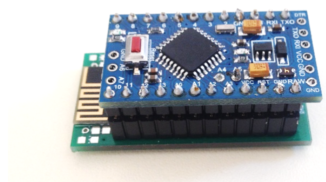

Figure 2. The Arduino node with the nRF24L01 transceiver.

The Arduino Pro Mini 3.3V developing board has the advantages of low cost, small size and reduced power consumption thanks to the reduced amount of components. The board includes an Atmel ATmega328P microcontroller. To transmit data, a surface mount version of the nRF24L01 transceiver was chosen. nRF24L01 is an ultra-low power component that uses the proprietary ESB protocol in the 2.4 GHz band, with data rates of 250 kbps, 1 Mbps and 2 Mbps. With this version of printed antenna, at 0 dBm signal power and 250 kbps, we achieved a transmission range of 250 m in line-of-sight [12]. This set up was also used in our test.

To build a compact wireless sensor node, a printed circuit board was developed to connect the Arduino Pro Mini board with the nRF24L01 transceiver. It also offers the possibility to attach different sensors to the microcontroller's pins. For our tests we acquired data from the sensors summarized in Table II. For each parameter, the data is acquired on 2 bytes. The power was provided by a 5 V power bank.

TABLE II. Used sensors for the Arduino node.

Measured Parameter	**Sensor**
Air Temperature	DHT21/AM2301
Air Humidity	
Soil Temperature	DALLAS DS18B20
Soil Humidity	DECAGON EC-5
Barometric Pressure	FREESCALE MPL115A1
Light Intensity	TEMT6000
Rain presence	Custom resistive sensor

C. LoPy node with LoRa

The energy saving algorithm with LoRa communication protocol was tested using an off-the-shelf solution, the developing board LoPy from Pycom, see Fig. 3. It uses the Semtech LoRa transceiver SX1276 in the 868 MHz band (Europe), with a maximum transmission power of +14 dBm. The data transmission range specified in the datasheet is up to 40 km. Additionally, it can also use the Wi-Fi and Bluetooth protocols [13].

Figure 3. The LoPy node with the PySense expansion board.

The producer also offers an expansion board with environmental parameters sensors, called PySense. The board provides a socket to attach additional sensors. The acquired parameters of interest are summarized in Table III. For each parameter there are also used 2 data bytes. The transmitted power level was set up at a maximum level, with a spreading factor of 12. The power was provided by a 5 V power bank.

TABLE III. USED SENSORS FOR THE LOPY NODE

Measured Parameter	**Sensor**
Air Temperature	PySense
Air Humidity	
Light Intensity	
Barometric Pressure	
Altitude	
Soil Temperature	DECAGON RT-1
Soil Humidity	DECAGON EC-5

III. DATA COMPRESSION ALGORITHM

To reduce the transmitted data of the acquired parameters, we have implemented an energy saving framework that includes data compression (a modified static Huffman algorithm), prediction (extrapolation prediction that exploits temporal correlation) and data aggregation (bit aggregation technique) [10].

The modified static Huffman algorithm reduces the data by coding the difference between the actual acquired value and the previous acquired value: $d_i = v_i - v_{i-1}$. Using the dictionary from Table IV, the transmitted data is defined as the concatenation of the binary representation of the calculated difference d_i and the representation s_i of the number of bits n_i used to code the calculated difference.

TABLE IV. THE HUFFMAN DICTIONARY.

n_i	s_i	d_i
0	00	0
1	01	-1,+1
2	100	-3,-2,2,3
3	101	-7,…,-4,+4,…,7
4	110	-15,…,-8,8,…,15
5	1110	-31,…,-16,16,…,31
6	11110	-63,…,-32,32,…,63
7	111110	-127,…,-64,64,…,127
8	1111110	-255,…,-128,128,…,255

To further reduce the data, the previous acquired value is replaced with the prediction of the actual acquired value from the previous three values:

$$\hat{v}_i = \frac{5}{4}v_{i-1} + \frac{1}{2}v_{i-2} - \frac{3}{4}v_{i-3}. \qquad (1)$$

The obtained data is finally encapsulated and transmitted on bytes. If each value is independently converted from bits to bytes, the compression gain could be lost. By applying the bit aggregation, firstly, the bits are merged, and then, they are divided in bytes which are further transmitted.

For example, if we have to transmit two uncompressed values such as air temperature and air humidity, which are coded on 12 bits each, the node will send 4 bytes. If we use the Huffman algorithm and extrapolation prediction, the temperature value could be coded on 3 bits and humidity value on 3 bits, resulting 2 bytes to transmit. Further, if bit aggregation technique is used, it results 6 bits, thereby only one byte should be transmitted.

The proposed framework was implemented in a software platform developed in Visual Basic environment. The data acquired by the presented wireless sensor nodes was introduced in a database, and the results obtained in terms of amount of bytes transmitted with and without compression were used to measure the current consumption for each node.

IV. EXPERIMENTAL RESULTS

The nodes were programmed to transmit the equivalent amount of data for the two cases: for the raw acquired data and for the energy saving framework implemented on raw data. Considering the microcontroller current consume significantly lower than the transceiver current consume, the node average current consumption and the node current consumption during data transmission were measured.

A. DASMote node with ZigBee

DASMote node with ZigBee protocol acquires 7 parameters and transmits 14 data bytes without data compression. The average current recorded in transmission is 27 mA, while the current spend on 58.45 ms transmission timeframe is 1.59 mAs. In Fig. 4 is captured the current consumption during data transmission with compression. The average current remains the same, while the current spend is reduced at 1.26 mAs because of a 46.45 ms transmission timeframe, thus resulting a ~20% energy improvement.

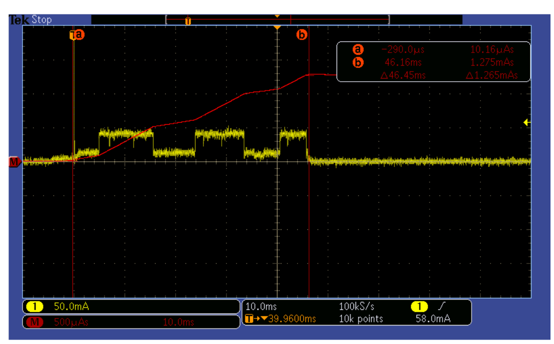

Figure 4. Current consumption of DASMote node.

B. Arduino node with Enhanced ShockBurst

For the Arduino node with the Enhanced ShockBurst protocol, there are acquired 7 parameters.

The node transmits 14 bytes without data compression, 2 bytes for each parameter. The average current in transmission is 6 mA, while the current spend on a 4.64 ms timeframe is 28.1 µAs. In Fig. 5 is captured the current consumption during data transmission with data compression. The current consumption is reduced at 26.29 µAs (4.34 ms transmission timeframe), resulting a ~7% energy improvement.

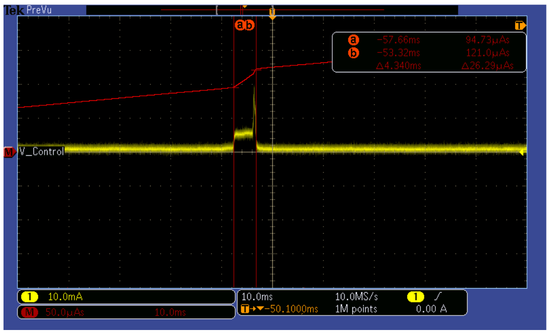

Figure 5. Current consumption of Arduino node

C. *LoPy node with LoRa*

The LoPy node with LoRa acquires also 7 parameters and transmits 14 data bytes without data compression. The average current in transmission is 154 mA, while the current spend on an 1190 ms timeframe is 183 mAs. Implementing the energy saving framework, the average current remains the same, while the current spend is reduced at 126 mAs. In this case, a ~31% energy improvement is obtained. In Fig. 6 is captured the current consumption during transmission with data compression.

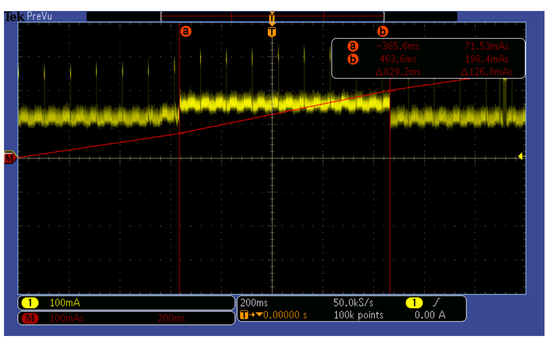

Figure 6. Current consumption of LoPy node.

V. Conclusions

In this paper, a study concerning the influence of data compression in energy consumption of wireless sensor nodes is presented. A comparison is made for nodes that uses the LoRa protocol for data transmission with nodes that uses other common data transmission protocols used in WSN, like ZigBee or Enhanced ShockBurst protocol

The advantages of LoRa long range data communication is reflected in a long data transmission timeframe. Implementing data compression algorithms we proved that the transmission timeframe is significantly reduced, resulting approximately 30% of lifetime improvement for data transmission. For the hardware platform tested with ZigBee communication protocol we recorded a 20% lifetime improvement, while from the hardware platform tested with Enhanced ShockBurst communication protocol we recorded only 7% lifetime improvement. The lifetime is prolongedat the protocols where the communication is achieved on a longer timeframe and higher range, and it decreases once with data transmission timeframe.

Acknowledgment

This work has been funded by University Politehnica of Bucharest, through the “Excellence Research Grants” Program, UPB – GEX 2017. Identifier: UPB- GEX2017, Ctr. No. 35/2017 (LoRa-WSN)”.

References

[1] International Electrotechnical Commission, “Internet of Things: Wireless Sensor Networks” IEC White Paper, 2014-11-01.

[2] Z Pei, Z Deng, B Yang, X Cheng, “Application-oriented wireless sensor network communication protocols and hardware platforms: A survey”, IEEE International Conference on Industrial Technology ICIT 2008., Chengdu, China, April 21-24, 2008.

[3] ZigBee Alliance, http://www.ZigBee.org/

[4] Nordic Semiconductor (http://www.nordicsemi.com/)

[5] U. Raza, P. Kulkarni, M. Sooriyabandara, “Low Power Wide Area Networks: An Overview”, IEEE Communications Surveys & Tutorials, Vol. 19, Issue 2, pp. 855–873, Jan. 2017.

[6] R. S. Sinha, W. Yiqiao, S.H. Hwang, “A survey on LPWA technology: LoRa and NB-IoT”, ICT Express. 3. 10.1016/j.icte.2017.03.004., March 2017

[7] LoRa Alliance (https://www.lora-alliance.org/)

[8] A. J. Wixted, P. Kinnaird, H. Larijani, et.al, „Evaluation of LoRa and LoRaWAN for wireless sensor networks”, SENSORS, 2016 IEEE, 30 Oct.-3 Nov. 2016.

[9] C.M. Sadler, and M. Martonosi, “Data compression algorithms for energy-constrained devices in delay tolerant networks”, 4th International Conference on Embedded Networked Sensor Systems (SenSys), pp. 265–278, 2006.

[10] D.I. Săcăleanu, L. A. Perişoară, R. Stoian, V. Lăzărescu, “A New Energy Saving Framework for Long Lasting Wireless Sensor Nodes”; 7th International Conference on New Technologies, Mobility and Security (NTMS2015), Paris, France, July 27-29, 2015.

[11] D. I. Săcăleanu, L. A. Perişoară, V. Lăzărescu, R. Stoian, „A New Multipurpose Wireless Sensor Node for Data Acquisition Systems”, 6th International Conference on Electronics, Computers and Artificial Intelligence (ECAI 2014), Bucharest, Romania, Vol. 6, No. 2, pp. 35-38, Oct. 23-25, 2014.

[12] D. I. Săcăleanu, L. A. Perişoară, E. Spătaru, R. Stoian, “Low-cost wireless sensor node with application in sports”, 23rd IEEE International Symposium for Design and Technology in Electronic Packaging (SIITME), Constanta, Romania, pp. 395-398, Oct. 26-29, 2017.

[13] https://pycom.io/