Chapter #

AN INTRA-VEHICULAR WIRELESS SENSOR NETWORK BASED ON ANDROID MOBILE DEVICES AND BLUETOOTH LOW ENERGY

José Augusto Afonso[1], Rita Baldaia da Costa e Silva [2], and João Luiz Afonso[3]

[1] *CMEMS-UMinho R&D Center, University of Minho, Guimarães, 4800-058, Portugal, e-mail: jose.afonso@dei.uminho.pt, phone: 351-253510184, fax: 351-253510189.*

[2] *CMEMS-UMinho R&D Center, University of Minho, Guimarães, 4800-058, Portugal, e-mail: a58677@dei.uminho.pt, phone: 351-253510190, fax: 351-253510189.*

[3] *Centro Algoritmi, University of Minho, Guimarães, 4800-058, Portugal, e-mail: jla@dei.uminho.pt, phone: 351-253510183, fax: 351-253510189.*

Abstract: This chapter presents the development and test of an intra-vehicular wireless sensor network (IVWSN), based on Bluetooth Low Energy (BLE), designed to present to the driver, in real-time, information collected from multiple sensors distributed inside of the car, using a human-machine interface (HMI) implemented on an Android smartphone. The architecture of the implemented BLE network is composed by the smartphone, which has the role of central station, and two BLE modules (peripheral stations) based on the CC2540 system-on-chip (SoC), which collect relevant sensor information from the battery system and the traction system of a plug-in electric car. Results based on an experimental performance evaluation of the wireless network show that the network is able to satisfy the application requirements, as long as the network parameters are properly configured taking into account the peculiarities of the BLE data transfer modes and the observed limitations of the BLE platform used in the implementation of the IVWSN.

Keywords: Android, Bluetooth Low Energy, Electric vehicle, Human-machine interface, Intra-vehicular networks, Mobile phone sensing, Performance evaluation, Wireless sensor networks.

1. INTRODUCTION

Modern vehicles are highly automated, with the main functionalities controlled by multiple microprocessor-based electronic control units (ECU) spread inside the vehicle, which collect data from sensors and communicate with actuators and data sinks. Currently, these devices are mostly connected through cables using a network technology called CAN (Controller Area Network) [1].

The increasing complexity of vehicles and the rise of the number of applications and devices that they encompass increase the quantity of cables required, which introduces challenges such as increased weight, limitations on the placement of sensors and the change of cables when necessary [2]. The replacement of cables by wireless links has the potential to reduce the weight of the vehicle, resulting in lower fuel consumption, increase the mobility and flexibility of the system, and decrease the assembly and maintenance costs.

Given these advantages, the use of wireless technologies was recently proposed to provide the required communication inside the vehicles, forming a new type of network called IVWSN (Intra-Vehicular Wireless Sensor Network) [3].

The transition to a wireless network must be gradual, starting with non-critical systems and sensors in areas not easily accessed with cables, such as inside the car tires. The flexibility and convenience provided by wireless networks also allow the installation of sensors on demand, for example, to measure the temperature or other physical variable on a given place of the car when desired. It also makes possible the integration of wearable sensors [4] to monitor the driver's physiological state and provide alerts when the user is not in condition to drive.

The choice of the wireless sensor node technology for IVWSNs must take into account the following requirements [2]: low cost, low energy consumption, low latency, high reliability and support for messages with different priorities.

As discussed in the next section, Bluetooth Low Energy (BLE) stands as the most promising wireless technology currently available on the market to fulfill these requirements. Another advantage of BLE over some of the competing technologies, such as ZigBee, is the native support on smartphones. These reasons motivated the choice of BLE for the development of the intra-vehicular system described in this chapter. Given the importance of reliability in the context of these networks, this chapter also presents an experimental evaluation of the packet error rate (PER) achieved with BLE under different conditions.

The system described in this chapter was designed, implemented and tested using a plug-in electric vehicle (Fig. 1) named CEPIUM (*Carro Elétrico Plug-In da Universidade do Minho - in Portuguese language*), which was developed by the Group of Energy and Power Electronics (GEPE) of University of Minho.

Figure 1. Electric vehicle used in the development of the IVWSN.

This car is a Volkswagen Polo where the internal combustion engine parts were replaced by an electric motor, batteries and the electronic circuits required for the conversion into an electric vehicle [5].

This chapter provides a revised and extended version of a conference paper [6]. The main contributions presented in this chapter are: 1) The development of a system for data collection (from real sensor devices inside an electric vehicle), wireless transmission (using BLE) and real-time presentation (on a smartphone installed in the vehicle cockpit); 2) A performance evaluation of BLE in the context of IVWSNs, with the comparison between two data transfer modes: notifications and indications.

2. RELATED WORK

There are several short-range wireless network technologies available currently in the market: IEEE 802.15.4/ZigBee [7], IEEE 802.11/Wi-Fi [8], Bluetooth, among others. Bluetooth Low Energy (BLE), which was introduced in the Bluetooth 4.0 [9] standard, was developed by the Bluetooth Special Interest Group (SIG) as a low-power solution for monitoring and control applications. BLE appeared in response to a need to increase the lifetime of wireless devices powered by batteries, such as fitness and healthcare devices, wireless computer mice and keyboards, among others.

At the physical layer, BLE implements the LE (Low Energy) controller, which is not directly compatible with the BR (Basic Rate)/EDR (Enhanced Data Rate) controllers used by classic Bluetooth. Both versions operate in the 2.4 GHz ISM (Industrial, Scientific and Medical) frequency band. The BLE connection interval used for data transmission can be configured from 7.5 ms to several seconds, depending on the balance between latency and energy consumption requirements of the target application.

In [10], Tsai et al. evaluate the performance of ZigBee inside a vehicle in different places, under varied scenarios, with the engine on and off. The authors analyze the use of the Received Signal Strength Indicator (RSSI) and Link Quality Indicator (LQI) to evaluate the link quality and propose a detection algorithm based on RSSI/LQI/error patterns and an adaptive strategy to increase the link goodput while improving the power consumption.

Ahmed et al. [11] analyze the characteristics and performance of some wireless network technologies in the context of intra-vehicular networks and investigate issues related to the replacement of cabling between sensors/ECUs by wireless links. The authors compares RFID (Passive), Bluetooth, IEEE 802.15.4/ZigBee and UWB (Ultra-Wideband) standards. Based on this analysis and the requirements of IVWSNs, such as support for short payloads with low overhead, low latency and low transceiver complexity, the authors concluded that the most suitable wireless technologies for IVWSNs, among the studied ones, were the IEEE 802.15.4 protocols and an emerging UWB proposal. However, at the time of this study Bluetooth Low Energy was not available yet.

Most available wireless technologies with applicability in the context of IVWSNs share the same spectrum (the 2.4 GHz ISM band) with IEEE 802.11/Wi-Fi and Bluetooth networks. Therefore, it is important to evaluate the performance of potential IVWSNs technologies under interference of these networks. Such evaluation was conducted by Lin et al. [12] for both states of the vehicle: stopped and in movement. Experimental results indicate that Bluetooth Low Energy outperforms ZigBee in the vehicle under Wi-Fi interference. These results can be explained by the use of frequency hopping spread spectrum (FHSS) by the former, which increases the robustness against fading and interference because the transmissions are spread along all the available 2.4 GHz band, whereas ZigBee transmissions are limited to a fixed channel.

The authors in [13] provide an experimental evaluation of the power consumption of BLE using CC2540 modules (the same modules used in this chapter). With one data transmission per second and a sleep current of 10 µA, an average current consumption of just 23.9 µA was obtained. Assuming the use of a common CR2032 coin cell battery with capacity of 230 mAh, it can be concluded that the module would be able to operate continuously for 400 days with the same battery. These results show that

BLE is a suitable technology for applications that require low cost and low power consumption.

In [14], Afonso et al. evaluate the performance of BLE using notifications in the context of body area networks (BAN) with multiple sensor nodes generating high data rate traffic. Results show that the BLE protocol is suitable to the task. Moreover, in comparison with an alternative enhanced IEEE 802.15.4-based MAC protocol [15], it is able to consume less energy and supports more sensor nodes due to its higher physical layer data rate.

Unlike [14], this chapter also analyzes the performance of BLE with indications, besides notifications, and in the context of intra-vehicular networks.

3. SYSTEM DEVELOPMENT

Please The BLE-based intra-vehicular network developed in this work is composed by two types of nodes: a central station (smartphone) placed near the driver at the car cockpit, and multiple peripheral stations (BLE modules) placed all around the car near the sensors, as shown in Fig. 2.

Figure 2. Placement of the nodes of the BLE network inside of the car.

For this prototype two peripheral stations were developed to monitor relevant variables associated to the operation of the electric vehicle: the battery system node, placed below the rear seats, and the traction system node, placed inside the hood. In the future, more peripheral stations can be added to the network as required.

These nodes collect sensor data from two electronic systems developed by GEPE: the battery system and the traction system. The battery system provides information related to the parameters of the battery and the electric charger [16], whereas the traction system provides data related to the parameters of the electric motor controller, the state of charge and the temperature of the motor. All data is collected from the respective electronic systems via UART (Universal Asynchronous Receiver/Transmitter), except for the motor temperature, which is collected using the ADC (Analog-to-Digital Converter) of the BLE module.

The smartphone plays both the roles of central station of the BLE network and human-machine interface (HMI) of the system. It receives the sensor data from the peripheral stations via BLE, converts the integer values contained in the data frames into real values according to the respective units of measurement, and presents the information to the user in real-time.

3.1 Embedded Software

The hardware of the peripheral stations is based on the CC2540EM evaluation module, from Texas Instruments (TI). This module integrates an 8051-based microcontroller and a BLE transceiver in the same chip, as well as a connector for an external antenna and auxiliary components. During the development phase, each CC2540EM module was connected to a SmartRF05EB evaluation board, for easy access to the I/O pins and the download of the developed code to the microcontroller of the CC2540 system-on-chip (SoC). In an early phase the CC2530EM module was also used for the central station, but it was replaced later by an Android smartphone for the development of the HMI. The development of the embedded software in the BLE modules was made using C language and the IAR Embedded Workbench for 8051 IDE (Integrated Development Environment). The SmartRF Packet Sniffer was used for system testing along with a CC2540 USB Dongle.

The first stage of the code development was the establishment of periodic data transfer from the peripheral stations to the central station. It was based on the use of two example projects supplied by TI along with the BLE stack, version BLE-CC2540-1.3.2, named SimpleBLECentral e SimpleBLEPeripheral, as well as an adaptation of the service “Send Data Service” developed in [17]. The data transfer was implemented using both notifications and indications.

For this purpose, it was necessary to implement a profile to provide this service in the peripheral station, which assumed the role of GATT (Generic Attribute Profile) server. The periodic event SBP_PERIODIC_EVT, which was configured to occur every 500 ms, calls the function PeriodicSendDataTask() to send the data from the sensors attached to that node, which are collected with the same period under the event UART_PERIODIC_EVT.

The service responsible for sending the data offers a profile with a single characteristic with maximum length of 20 bytes and a descriptor (CCC -

Client Characteristic Configuration). If the network is configured for indications, the peripheral station waits for an acknowledgment (ACK) for each data frame sent.

It was necessary to make changes on the original code to allow the communication using indications. In the peripheral station, it was needed to create a characteristic which assumes the value GATT_PROP_INDICATE and register the CCC descriptor with the value 0x0002. It was also necessary to change the data structure to attHandleValueInd_t and, finally, to use the function GATT_Indication to send the data frames to the central station. In the central station, it was only necessary to add the function ATT_HandleValueCfm to send the acknowledgement for each frame correctly received.

3.2 Communication Protocol

In order to regulate the transfer of data from either the battery system or the traction system and the respective BLE module (peripheral station) via UART, a polling protocol was implemented, with each BLE module as master and the respective electronic system as slave. The master polls the slave with a period defined by the event UART_PERIODIC_EVT and waits for the response. The poll frame contains a single field (1 byte) which indicates the data content requested from the attached electronic system. The data frame sent by the battery system is composed by the system address, frame type, frame length and the samples from its 6 sensors: grid voltage, grid current, battery voltage, battery current, bus voltage and temperature. Likewise, the data frame sent by the traction system is composed by the system address, frame type, frame length and the samples from its 6 sensors: controller state, controller voltage, controller current, state of charge, controller power and controller temperature. An error message frame is sent if the electronic system is not able to send the requested data frame.

3.3 Smartphone Application

The Android [18] application was developed for Android 6.0 using a Motorola Moto G (2nd gen) mobile device and the Android Studio 2.1 IDE. It was designed to provide a simple and flexible human-machine interface, offering access to the data collected from the car sensors in an organized way. For this purpose, the application was created using mainly fragments, simplifying the reuse of components of the user interface.

The developed application uses four activities. The main activity (starts menu) serves as the initial user interface of the application and is responsible for the management of fragments. It provides access to the battery system screen, the traction system screen and the weather information screen. This activity uses a navigation drawer that manages the different fragments in order to guarantee their visibility. This panel can be accessed with a click on the Android hamburger icon or with a swipe from left to right.

The second activity is responsible for the BLE device discovery, connection establishment and activation of the reception of sensor data from the peripheral stations using notifications. The BluetoothLeService file is responsible for managing these features. Besides playing the role of central station, the smartphone also acts as the ATT (Attribute Protocol) client, accessing data from the peripheral stations, which assume the role of ATT servers, maintaining a set of attributes and storing information managed by the GATT protocol. In order to use Bluetooth on an Android smartphone, it is necessary to create the BluetoothAdapter object. For device discovery, the startLeScan () method is called to start searching for active BLE devices, whereas the stopLeScan () is used to stop this search. By default in the Google API, this scan process stops after 4 seconds, which is sufficient to find the peripheral stations.

The third activity implemented in the smartphone application is responsible for the sensor data presentation. After the notifications are activated, this activity transfers the data frames received from the peripheral stations to the DataHandler class, to extract and process the sensor data contained in the fields of the data frames in real time. The periodic data reception associated to this task is managed by a background service, allowing the user to perform other operations simultaneously.

The last implemented activity provides weather information (current weather and forecast for the next 3 days), based on the current location of the mobile device. This weather information is collected from the online service OpenWeatherMap using its JSON (JavaScript Object Notation) API (Application Program Interface).

4. RESULTS AND DISCUSSION

4.1 Overall System

The goal of this first test was to evaluate the correct operation of all parts of the developed system, from the collection of data from the sensors of the electronic system using the developed communication protocol to the presentation of this information on the smartphone. Table 1 presents the expected values for the different sensors of the battery system, whereas Fig. 3 shows an example of the presentation of this sensor values on the screen of the smartphone (which was configured with limits larger than the specified on the table). The same test was performed for the traction system, with similar satisfactory results.

Table 1. Sensor values for the battery system.

Sensor Data	Minimum Value	Nominal Value	Maximum Value	Unit
Grid voltage	210	230	240	V
Grid current	-	16	20	A
Battery voltage	-	300	360	V
Battery current	-	10	13	A
Bus voltage	-	430	450	V
Temperature	-	-	100	ºC

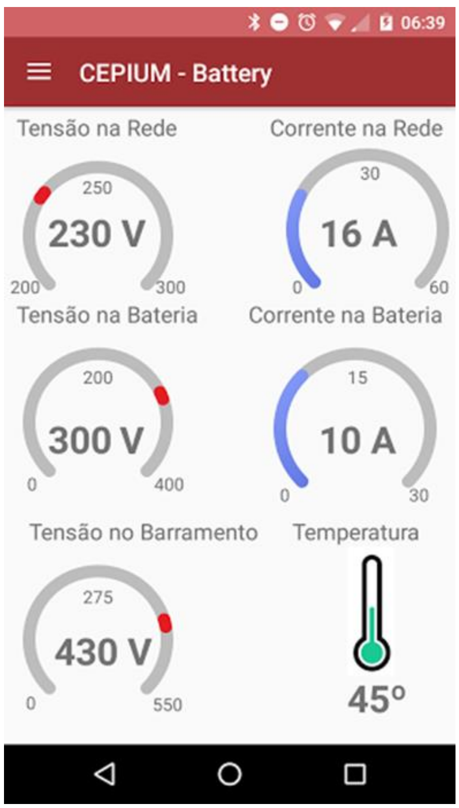

Figure 3. App screen for presentation of the battery system sensor values.

4.2 Setup for the BLE Performance Tests

The following experimental tests evaluate the BLE network reliability by measuring the packet error rate (PER) after the transmission of 1000 data frames from a peripheral station to a central station using CC2540EM modules and either notifications or indications. A packet is only considered lost if the corresponding data frame is not successfully delivered to the central station, either in the first attempt or after retransmissions (in the case of indications).

After a BLE connection between the central station (master) and a peripheral station (slave) is established, time is divided into connection events [9]. Each connection event starts with a packet from the master (poll frame), which is followed by a packet from the slave (data frame). Depending on the configuration, the connection event may then terminate or continue with the transmission of subsequent packets, alternating from the master and the slave. If the slave latency parameter is equal to zero, the interval between active connection events is equal to the connection interval.

The tests were performed using four different transmitter (TX) power levels: -23 dBm, -6 dBm, 0 dBm e 4 dBm. These tests used the maximum allowed payload length (20 bytes) and a single data frame per notification or indication. The period between connection events is a BLE parameter called connection interval (T_{ci}) and the period of generation of data frames is a TI stack parameter called the data send period (T_{ds}) in this chapter. All tests with notifications were made with $T_{ds} = T_{ci}$, whereas in the tests with indications the value of T_{ds} was a multiple of T_{ci}.

4.3 Performance Tests Inside of the Car

These tests were performed with the goal to evaluate the communication reliability inside of the vehicle, in order to assess if its metallic parts might have effect on the strength of the signal and if the interference from the electric motor might cause errors. This tests were performed using notifications and the minimum TX power (-23 dBm), with the central station placed at the car cockpit and the peripheral station at different places on the vehicle. For the first test, the peripheral station was placed near the electric motor, with the hood closed, while the motor was accelerated. No packet errors were registered during this test. The same test was repeated with the peripheral station near the battery system, the front lights, the rear lights and the four tires, with the same result

(PER = 0). Therefore, it was not necessary to make further tests with higher TX power levels.

4.4 Tests at Short Distance

The goal of the following tests is to determine the minimum value for the connection interval and the data send interval from which communications are not affected by errors that are caused exclusively due to limitations of the TI BLE platform implementation, both for notifications and for indications. For this purpose, the tests were made in a controlled setup to exclude any other possible causes of packet errors: low path loss (short distance of 1 m between the stations), no obstructions and no interference from other sources such as Wi-Fi networks.

Table 2 presents the results with notifications using TX power of -23 dBm. The same results were obtained with higher power levels. No packet errors were registered for T_{ci} equal to 10 ms and above. The Bluetooth specifications state that the connection interval may be configured to any multiple of 1.25 ms in the range of 7.5 ms to 4.0 s. However, the tested BLE platform does not perform as expected with T_{ci} = 7.5 ms.

Table 2. PER with notifications at distance of 1 m.

Connection Interval (ms)	PER (%)
7.5	78
10	0

The same test was repeated using indications instead of notifications. In this case, besides defining the value of the connection interval, it is also necessary to specify the value of the data send period as a multiple of the former. For each data frame sent it is necessary to wait for the reception of an ACK frame in the next connection event. When the ACK is not received the BLE stack retransmits the data frame at each subsequent connection interval until the ACK is received. If the data send period is exceeded while waiting for the ACK, the peripheral station blocks the transmission of the next data packet, causing its loss. Therefore, for proper operation of indications, it is necessary to make T_{ds} at least twice the value of T_{ci}, even when no retransmissions are required due to the absence of channel errors. This point was confirmed through tests using $T_{ds} = T_{ci}$, with connection intervals ranging from 7.5 ms to 25 ms, which resulted in PER values ranging from 53% to 55%.

Further tests were made using $T_{ds} = 2\ T_{ci}$ for different values of the connection interval. Table 3 shows the results obtained with -23 dBm. The same results were obtained with -6 dBm, 0 dBm e 4 dBm. Although, theoretically, configuring $T_{ds} = 2\ T_{ci}$ should be sufficient to obtain PER = 0, since the occurrence of channel errors (and consequently, the need for retransmissions) was excluded by the way the test was set up, these results show that the tested BLE platform introduces errors for low values of connection interval. Based on these results, it can be concluded that the minimum connection interval with indications recommended in this case is 100 ms. Moreover, to allocate the required connection events for retransmissions in real world scenarios susceptible to channel errors, it is necessary to make $T_{ds} > 2\ T_{ci}$.

Table 3. PER with indications at distance of 1 m when $T_{ds} = 2\ T_{ci}$.

Connection Interval (ms)	**Data Send Period (ms)**	**PER (%)**
7.5	15	25
10	20	17
15	30	11
20	40	7
25	50	7
50	100	3
60	120	3
100	200	0

To accommodate at least one retransmission with indications, it is necessary to make the data send period at least three times higher than the connection interval. Given these results, we conclude that for a reliable operation with the used BLE platform and indications, the minimum connection interval should be 100 ms and the minimum data send period should be 300 ms.

These results also show that the maximum possible throughput with indications in scenarios without channel errors is much lower than with notifications, because the minimum recommended data send period is much higher. Moreover, it is only possible to transmit one indication per connection event, whereas the same platform allows the transmission of up to three notifications in the same event, as verified in [14].

Nevertheless, for the purpose of the intra-vehicular network presented in this chapter, the platform may operate with either notifications or indications, since it was configured to send a single data frame each time with a data send period of 500 ms.

4.5 Tests at Larger Distance

The final test aims to evaluate the influence of the TX power and the connection interval, using notifications, on the data frame transmission reliability in a scenario where the stations are separated by a larger distance (10 m) and with obstructions. Table 4 presents the results for this test. The minimum value for the connection interval presented in the table was 25 ms because it was not possible to establish connection with lower values, which means that the distance also has influence on the recommended minimum connection interval.

Table 4. PER with notifications at 10 m for different TX power values.

Data Send Period (ms)	**PER (%) -23 dBm**	**PER (%) -6 dBm**	**PER (%) 0 dBm**
25	100	30	0
33.75	88	0	0
50	76	0	0
60	50	0	0
80	48	0	0
100	43	0	0

These results show that the TX power has strong influence on the BLE transmission reliability for larger distances and that the minimum TX power is not adequate in this case. With -6 dBm it was possible to achieve error-free communication using connection intervals starting from 33.75 ms and with 0 dBm from 25 ms.

5. CONCLUSION AND FUTURE WORK

The main objectives of this work were the development of a BLE network to collect sensor data inside a vehicle, the development of an

Android app to present the collected information to the driver in real-time and the execution of experimental tests to evaluate the reliability of the BLE network with both notifications and indications.

Based on the literature review and obtained experimental results, it can be concluded that the characteristics and performance of Bluetooth Low Energy make it one of the most promising technologies for the deployment of intra-vehicular wireless sensor networks when compared to the existing alternatives. Nevertheless, eventual limitations of the used BLE platform should be taken into account during the implementation and configuration of the network.

In the future, the developed system will evolve to connect the smartphone application to an IoT (Internet of Things) [19] architecture composed by an online database service, a client application and a residential WSN (Wireless Sensor Network) [20]. Together with the integration of new sensors and actuators into the system, this evolution will allow the monitoring and control of parameters of the vehicle anywhere in the world through the Internet, enabling the provision of several new features, such as the manual/automatic remote control of the battery charging current according to the user's preferences.

ACKNOWLEDGMENT

This work is supported by FCT with the reference project UID/EEA/04436/2013, COMPETE 2020 with the code POCI-01-0145-FEDER-006941.

REFERENCES

1. M. Di Natale, H. Zeng, P. Giusto, and A. Ghosal, "Understanding and using the controller area network communication protocol: theory and practice," Springer Science & Business Media, 2012.
2. J. Lin, T. Talty, and O. Tonguz, "On the potential of Bluetooth Low Energy technology for vehicular applications," IEEE Communications Magazine, vol. 53, no. 1, pp. 267-275, January 2015.
3. L.M. Borges, F.J. Velez, and A.S. Lebres, "Survey on the Characterization and Classifica-tion of Wireless Sensor Network Applications," IEEE Communications Surveys & Tutorials, vol. 16, no. 4, pp. 1860-1890, 2014.
4. A. Pantelopoulos and N.G. Bourbakis, "A survey on wearable sensor-based systems for health monitoring and prognosis," IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews, vol. 40, no. 1, pp. 1-12, January 2010.
5. D. Pedrosa, V. Monteiro, H. Gonçalves, J.S. Martins, and J.L. Afonso, "A case study on the conversion of an internal combustion engine vehicle into an electric vehicle," in IEEE Vehicle Power and Propulsion Conference (VPPC 2014), October 2014.

6. Rita B.C. Silva, Jose A. Afonso, and Joao L. Afonso, "Development and Test of an Intra-Vehicular Network based on Bluetooth Low Energy," Lecture Notes in Engineering and Computer Science: Proceedings of The World Congress on Engineering 2017, 5-7 July, 2017, London, U.K., pp. 508-512.
7. P. Castro, J.L. Afonso, and J.A. Afonso, "A Low-Cost ZigBee-based Wireless Industrial Automation System", in 12th Portuguese Conference on Automatic Control (CONTROLO 2016), September 2016. pp. 739-749.
8. IEEE Std 802.11-2016, "IEEE Standard for Information technology—Telecommunications and information exchange between systems Local and metropolitan area networks—Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," 2016.
9. Bluetooth SIG, "Specification of the Bluetooth System. Master Table of Contents & Compliance Requirements," Version 4.0 [Vol 0], June 2010.
10. H.M. Tsai, C. Saraydar, T. Talty, M. Ames, A. Macdonald, and O.K. Tonguz, "ZigBee-based intra-car wireless sensor network," in IEEE International Conference on Communications, 2007, pp. 3965–3971.
11. M. Ahmed, C.U. Saraydar, T. Elbatt, J. Yin, T. Talty, and M. Ames, "Intra-vehicular Wireless Networks," in IEEE Global Communications Conference (GLOBECOM), 2007, pp. 1–9.
12. J.R. Lin, T. Talty, and O.K. Tonguz, "An empirical performance study of Intra-vehicular Wireless Sensor Networks under WiFi and Bluetooth interference," in IEEE Global Communications Conference (GLOBECOM), 2013, pp. 581–586.
13. S. Kamath and J. Lindh, "Measuring Bluetooth Low Energy Power Consumption," Application Note AN092, Texas Instruments, pp. 1–24, 2012.
14. J.A. Afonso, A.J.F. Maio, and R. Simoes, "Performance Evaluation of Bluetooth Low Energy for High Data Rate Body Area Networks," Wireless Personal Communications, vol. 90, no. 1, pp. 121–141, September 2016.
15. J.A. Afonso, H.D. Silva, P. Macedo, and L. A. Rocha, "An Enhanced Reservation-Based MAC Protocol for IEEE 802.15.4 Networks," Sensors, vol. 11, no. 4, pp. 3852–3873, April 2011.
16. V. Monteiro, J. P. Carmo, J. G. Pinto, and J. L. Afonso, "A Flexible Infrastructure for Dynamic Power Control of Electric Vehicle Battery Chargers," IEEE Transactions on Vehicular Technology, vol. 65, no. 6, pp.4535-4547, June 2016.
17. A.F. Maio and J.A. Afonso, "Wireless Cycling Posture Monitoring Based on Smartphones and Bluetooth Low Energy", Lecture Notes in Engineering and Computer Science: Proceedings of The World Congress on Engineering 2015, 1-3 July, 2015, London, U.K., pp. 653-657 .
18. N. Gandhewar and R. Sheikh, "Google Android : An Emerging Software Platform For Mobile Devices," Int. J. Comput. Sci. Eng., no. 12, pp. 12–17, 2010.
19. R. Want, B.N. Schilit, and S. Jenson, "Enabling the Internet of Things," IEEE Computer, vol. 48, no. 1, pp. 28-35, 2015.
20. M. Collotta and G. Pau, "Bluetooth for Internet of Things : A fuzzy approach to improve power management in smart homes," Computers & Electrical Engineering, vol. 44, pp. 137-152, May 2015.