Designing a Gateway Device for Internet of Things Applications

Deniz Taşkın[1], Mustafa Çetingöz[1*]

[1] Computer Engineering, Engineering Faculty, Trakya University, Edirne, Turkey
* Corresponding author's e-mail: mustafacetingoz@trakya.edu.tr

ABSTRACT

BLE (Bluetooth Low Energy) technology is an ideal tool for implementing IoT (Internet of Things) applications, as it consumes little power and provides wireless communication. There are many applications using this technology already available on the market. These products are classified as personal applications while using the BLE-supported tablets or smartphones for transmitting data and are not institutional. There are numerous studies showing that the number of IoT sensors and devices will increase rapidly. However, personal approaches cannot manage a large number of devices. In this study, a BLE gateway device which interconnects BLE and WiFi networks was developed. This device is also capable of transmitting IoT node information to a server or cloud on the Internet. The proposed design uses only BLE and WiFi stack chips and does not require an additional microcontroller unit for running gateway applications. This makes the design cost-effective and power efficient, which are important for the IoT environment.

Keywords: Bluetooth low energy, Internet of things, gateway.

INTRODUCTION

The Internet of Things is a global network of smart objects, such as specially designed devices, clothes, and sensors that interact with each other through the Internet protocols [4]. Alternatively, it can be defined as the communication and information sharing of uniquely addressable devices using standard Internet communication protocols. First mentioned in 1999 by Kevin Ashton to recount the benefits of RFID technology, it then became a global system standard [8].

There is a continuous increase in the Internet-connected devices such as health monitors, sensors, industrial automation devices, vehicles, and home appliances. The number of these devices has significantly increased compared to the world population, and by 2020 it is predicted that the number of devices connected to the internet will reach 50 billion [1]. It is common to see the smartphone applications used for transferring data from sensors to the central device [3, 10, 21]. However, large-scale approaches, like hospitals or smart cities, require more than standalone smartphone applications.

In general, BLE and ZigBee technologies are used for the Internet of Things and wireless personal area networks [1]. BLE, which was first announced in 2010, offers an additional communication topology based on the broadcasting method. This method is based on two basic roles, broadcaster and observer. The information can be transferred to observer devices without a regular peer to peer connection. This differs from peer to peer network topology, as this method enables to collect data from multiple broadcaster nodes. The technology aims at transferring a small amount of data continuously for over a year with a coin battery [16].

There are two critical points regarding the data transfer from the sensor nodes used in the Internet of Things. In the first, the data obtained from sensors is collected with minimal energy consumption; and the other is a cost-effective transfer of the data to the central device over the Internet.

Power consumption is one of the most important issues in short-range wireless communication technologies. A device with high power consumption is not suitable for mobile applications and must be used with a constant power supply

instead of a battery. While comparing Zigbee and BLE power consumption, it is clear BLE technology offers the lowest energy consumption [2, 13].

In order to provide cost-effective data transfer, it is important to use the existing network infrastructure. WiFi technology is a dominant actor in computer networks, but it is difficult to combine with IoT solutions [12].

In this study, a gateway device was designed to incorporate IoT sensor nodes into the existing WiFi network. The design enables to transfer the data obtained from sensors to the Internet or cloud. Such an approach acts as a bridge and enables complex structures that consist of hundreds of sensors to be built.

RELATED WORKS

Boualouache et al. proposed a BLE-based system for IoT applications [2]. This system uses a mobile phone as a bridge device, similar to others. The IoT nodes are designed using the Arduino platform. The phones collect the data from these nodes and push them to the Internet.

In the study proposed by Laine, Lee, and Suk, Zigbee is used for instant data transfer in health systems. While Zigbee cannot meet the required bandwidth, a Bluetooth-based gateway solution is offered [11]. In the system, the information generated from the sensors is transferred to the node point with Zigbee. This information is collected by the node point and transferred to the patient's mobile gateway device via Bluetooth. In the patient's mobile gateway device, this information is pre-processed and then transferred to a server on the internet via WiFi, such as 3G or 4G network.

Zachariah et al. addressed the gateway problem for the IoT and proposed a software-based solution. A smart mobile device is controlled by a mobile application to act as a gateway. The data received by the mobile smartphone is transmitted via WiFi to the Internet or to the cloud [22].

The system recommended by Hawelkar and Tamhanani combines the technologies such as BLE and ZigBee for short-range wireless communication on a single gateway device [9]. The architecture of the design is Linux-based and uses 32-bit MCU (Main Control Unit). It also provides an Internet connection for both protocols.

A study by Giuseppe De Maso-Gentile et al. aimed at developing a BLE gateway device for monitoring the battery management of electric vehicles. In this study, an Arduino board and BLE system equipped with a controller area network interface was utilized and a smartphone application was used for receiving information. The gateway comprised the Bluno board, Arduino Bluetooth 4.0 module, and transferred information read from a car battery management system [6].

PROPOSED SYSTEM, IOT GATEWAY

In the previous section, the studies on transferring data from the IoT sensor nodes were examined [2, 22]. Although in all studies the data is transferred from sensor nodes to the Internet or cloud server, they are generally classified as personal applications because a personal mobile smartphone is used. On the other hand, there are some studies which use an additional MCU on the gateway device and an application runs on this processor [6, 17, 20]. The designs in these studies are costly due to extra MCU, CPU, hardware, or operating system usage and are not suitable for the IoT applications requiring low power consumption.
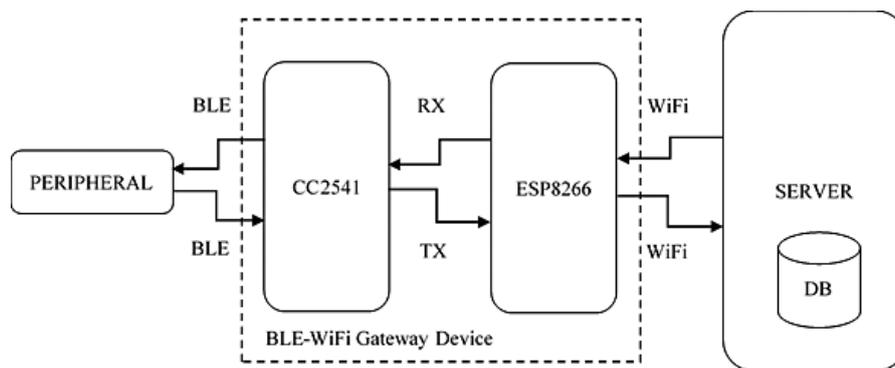

Fig. 1. Block diagram of IoT gateway device

In this section, we will discuss the created gateway device and consider the design problems in previous studies. The device has two main parts, one side BLE and the other side WiFi, which are responsible for transmitting the data received from the sensors to a server on the Internet via Bluetooth low energy. The device is primarily designed to combine two different wireless environments using a BLE and a WiFi module. This simple design makes our IoT gateway device cost-effective.

SYSTEM ARCHITECTURE

The IoT gateway device in the observer mode is responsible for receiving and transmitting packets which are broadcast by a sensor or a sensor group. These BLE advertising packets will be transferred to a server via WiFi through a web service. The block diagram of the proposed IoT gateway device is shown in Figure 1.

The chips used in BLE and wireless modules are both single device configuration, and therefore the recommended device is functionalized without the need for an extra processor. Advertising packets are received by the IoT gateway device's BLE and are sent as UART (Universal Asynchronous Receiver Transmitter) messages to the wireless module, which is physically connected via the serial input/output port. While the wireless module is always connected to a WiFi network, the received messages can be transferred to the internet or cloud server using web service method. A flow diagram of the entire communication is shown in Figure 2.

BLE MODULE

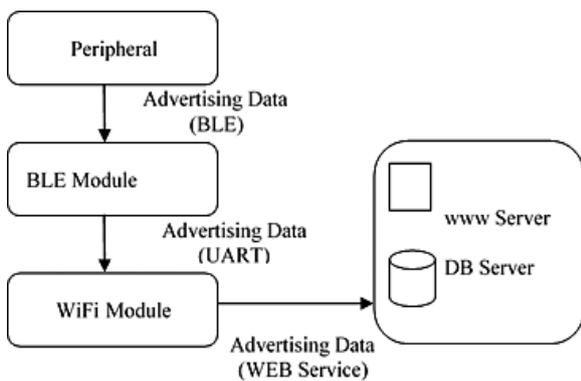The BLE module consists of a low energy wireless radio and its software infrastructure

Fig. 2. Flow of advertising data

(stack). The low cost and power consumption of the BLE module played an important role in the decision. A module with a single device configuration that does not require the use of an external processor was searched for. Using these criteria, the CC254x platform offered by Texas Instruments, US was preferred [14].

Two different stack and application configurations are supported on the CC254X platform, a single device configuration, and network processor configuration. The layers for both configurations are shown in Figure 3. The BLE devices consist of three basic blocks: application, host, and controller. The block allowed to communicate with the Bluetooth protocol stack, including this specific use case, is the application block, as the name implies.

The host block contains the top layers of the stack, and the controller block includes the bottom layers of the stack and radio communication. In addition, there is a host controller interface (HCI) that provides communication between the host block and the controller block to ensure the different manufacturers' products are working with each other [7].

The single device configuration is the simplest and most common configuration type. The controller, the host, the profiles, and the application are combined on a single chip, with low cost and low power consumption. In the network processor configuration, the controller and host layers are run together on the same CC254x chip, while profiles and applications are run on separate external processors. Application and profiles communicate with the CC254x via developer special HCI commands using the SPI, UART interface, and virtual UART over USB. This configuration is ideal for the situations where the application is run on an external microcontroller or a PC [18].

A BLE device can have four basic modes: master, slave, advertising, and scanning. The pairing process happens before communication can occur in the master and slave device modes. It is sufficient for the sender to be in the advertising mode and the receiver in the scanning mode when the devices send small capacity status information periodically without pairing.

In Bluetooth low energy technology, the communication with the outside world takes place in two ways, broadcasting and connection, each via the generic access profile (GAP). The broadcast and connection mechanisms have specific advantages, disadvantages, and limitations.
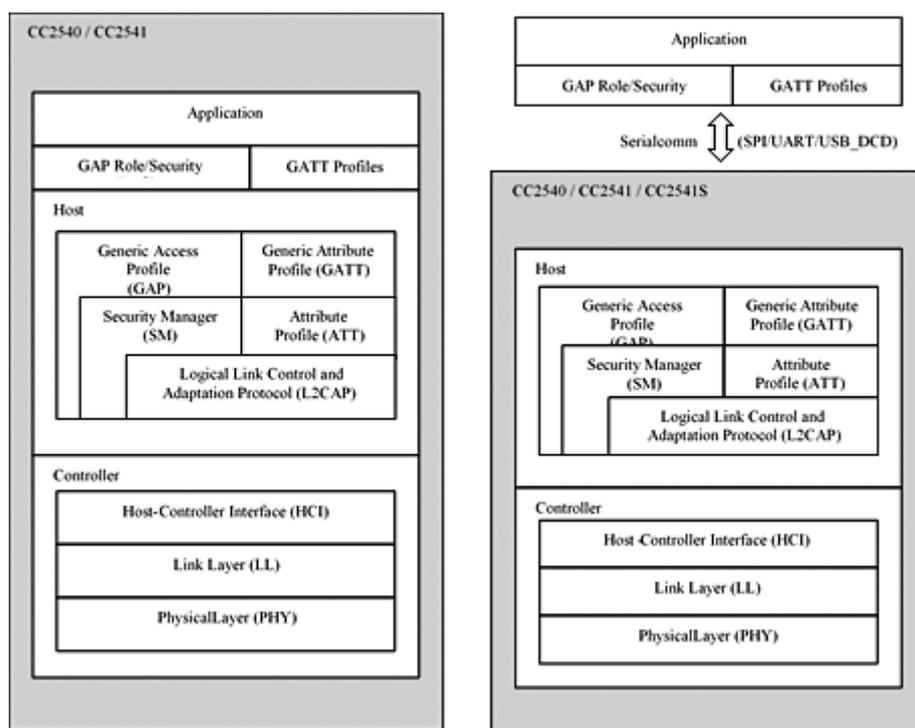
Fig. 3. BLE protocol layers, left : single device configuration, right : network processor configuration

In the broadcast mode, the data is transmitted in one direction to the scanner or the receiver, which may receive the data in the listening interval without a connection. There are two distinct role definitions, broadcaster and observer. The broadcast role is based on the delivery of connectionless advertising packets periodically to any device that wishes to receive them. The role of observer is based on scanning the predetermined frequencies to receive advertising packets without a connection. The broadcast role provides an advantage for the situations where the data is transmitted to more than one peripheral at a time [19].

The broadcast role is easy and useful when the size of data to be transmitted is small and sent to multiple peripherals. However, when compared with the usual connection method there are serious weaknesses, such as no security or special provisions. This means any observer can receive the data sent from the device and is therefore not suitable for the data requiring security or confidentiality. The devices in the peripheral send advertising packets (ADV_IND) over three advertising channels (37, 38 and 39) [5]. Such packet transmissions are kept as short as 31 bytes, depending on the load of the application. In addition to the required preamble, the packet header, MAC address, and other provisioning information are included.

The BLE advertising packet containing the sensor data was selected as a floating-point 4-byte data type that can be a positive/negative, integer, or decimal value. The peripheral identification information is also defined as 4 bytes due to the number of peripherals in the environment. The BLE peripheral also sends the information about the battery level. The peripheral advertising data, which is shown in Figure 4 as 10 bytes in total, includes 4-byte peripheral sensor value information, 4-byte peripheral identification information, 1-byte peripheral battery level, and 1-byte peripheral data length information.

On the BLE module component of the IoT gateway device, there is an application running that transforms the data sent by the peripheral unit into a UART package. A 1-byte version of the information for the application running on the BLE side of the IoT gateway device is added to the advertising data and transferred via the UART to the wireless module. The sample UART advertising data, which is 11 bytes in total, is shown in Figure 5.

WIRELESS MODULE

The TCP/IP stack and the WiFi chip will be referred to as the wireless module. While selecting the wireless module, cost effectiveness and
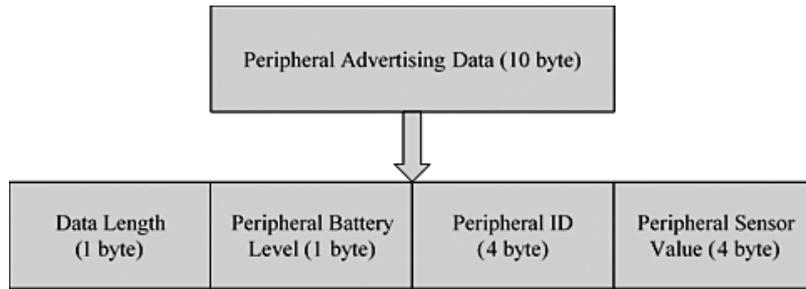
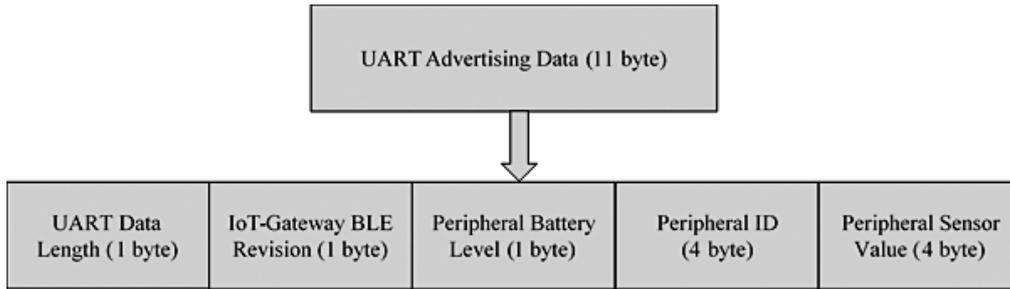**Fig. 4.** Format of peripheral advertising data

Fig. 5. UART advertising data format

low power consumption were primary concerns, along with offering programmability without the need for an additional processor. The ESP8266 module by Espressif, which has these features, was selected.

The ESP8266 is a complete and independent module that is easy to find on the market at a low cost, and acts as a secure and self-contained WiFi networking solution. It can be used to host the application or load WiFi network functions from another application processor. If the application is hosted, it can boot directly from an external flash memory, and also has integrated caching to improve system performance.

The entire solution was designed with minimal space required and an external circuit. It is an integrated WiFi chip that incorporates antenna switches, a power amplifier, low noise receiver amplifier, filters, and power management modules. The ESP8266 can be integrated with external sensors and other application-specific devices, usually via general purpose input/output (GPIO) interface. A 3.3 V and low power consumption are key features it has to offer. In addition, there is an application in the module for web service calling and channel multiplexing [15].

The data received by the UART is transferred to the server via web service methods through the wireless module component of the gateway device. However, the 1-byte length IoT gateway device battery level information, 1-byte length IoT gateway device wireless revision information, and 2-byte length IoT gateway device identification information were added to the UART advertising data format. A total of 15 bytes of web service advertising data is shown in Figure 6.

EXPERIMENTAL WORK

For the performance evaluation of the proposed IoT gateway device, a proper test environment was created and the software implemented. Figure 7 shows a sample sensor node with two sensors. In this sensor node BH1750 and MCP9808 was used. The BH1750 is a calibrated digital light sensor that measures the incident light intensity, and MCP9808 is used to measure temperature. Both sensors use the I2C protocol provided by CC2541 and can be increased by connecting to each other as needed.

Figure 8 shows the sample prototype of the designed IoT gate device. This device accesses the data from the sensor nodes via the CC2541 module and wirelessly transfers this data to the internet via the WiFi module.

A testing infrastructure was created to measure the performance of the proposed IoT gateway device. The web and database server was setup for data transfer. Then, a test table was
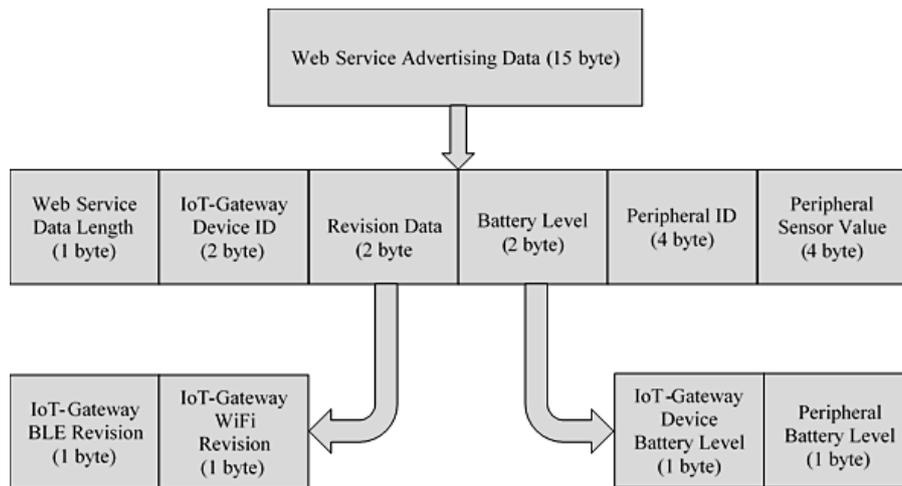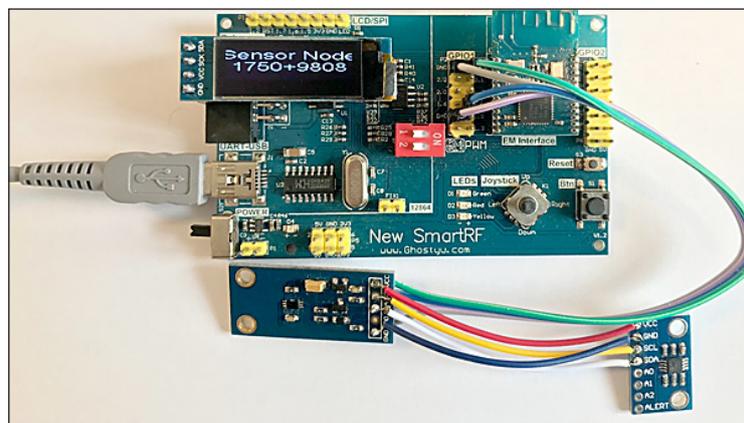
Fig. 6. Web service advertising data format

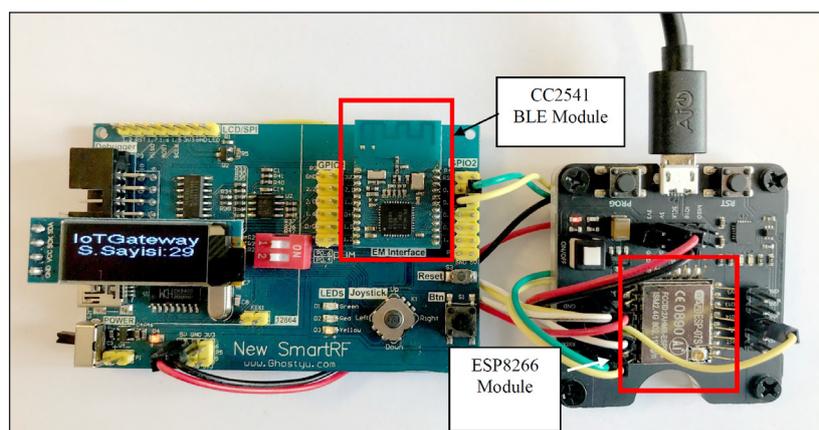**Fig. 7.** Sample sensor node

Fig. 8. Prototype of designed IoT gateway device

created in the database which includes identification information (ID), reader information (Reader_ID), raw data information (raw_data) and insertion time (insert_time). In order to prepare the test data set, a 15-byte long web service format was filled with random numbers. Although the data to be sent was 15 bytes long, the value reached 115 bytes while sending via the wireless module. These extra values were the server address, the PHP script, the values in the PHP request, the address of the requesting client, and other special characters.

The tests were performed on two different networks, LAN and WAN. As the name implies,

the LAN is on the same network as the server and the WAN is accessed over the internet. As seen in Figure 9, for WAN, the transmission of 100 packets was 3.44 per second, and for 1000 packets, 3.09 per second. When the LAN data is examined, 100 packets were sent at 4.16 per second, and for 1000 packets, 4.04 per second.

In order to send the advertising packets transmitted from the peripherals through the designed IoT gateway device seamlessly, it is necessary to configure the packet transmission interval and the number of peripheral devices. However, hop count, network capacity, and network traffic conditions between LAN or WAN and the servers should be considered.

Our second aim was to investigate the effect of advertising packet length. In this test, the number of packets sent was 1000 units and the initial packet length value was 128-bytes. The second packet length selected was 256-bytes. The results are recorded for 16 values up to 2048, increasing by 128-bytes each time. The values obtained for LAN and WAN are shown in Figure 10.

In this system, as the packet size increases, the sending speed decreases. Therefore, setting the transmitting packet at a lower size seems rational for the situations where the data should be sent at higher speeds. However, due to the restrictions of the wireless module, it is not possible to send a packet length of more than 2048-bytes. In this study, the packet size of web service advertising data is determined as 15 bytes, as the system has reached the appropriate response time for the submission of data.
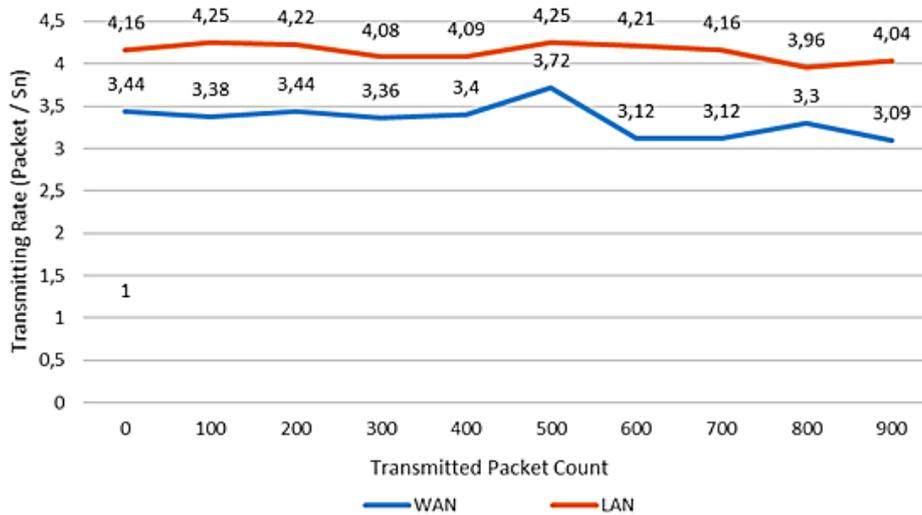

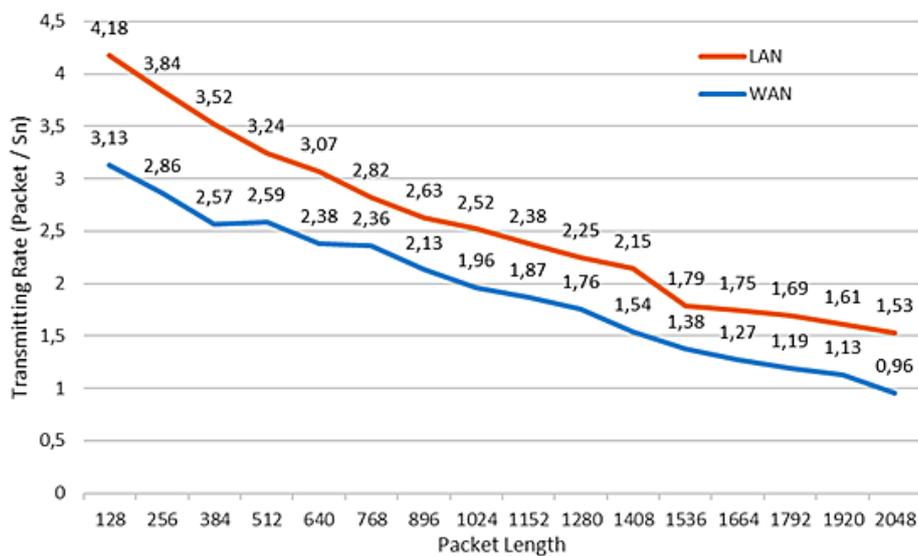**Fig. 9.** Transmission rates: LAN, WAN vs. packet count

Fig. 10. The effect of packet length vs transmitting rate

RESULTS AND CONCLUSION

In this study, we propose a gateway device that uses two different wireless environments through a common protocol by combining the ESP8266 and CC2541 chips. The proposed device is responsible for receiving the data from a sensor or sensor group using BLE and sending it to a server on the Internet or cloud via WiFi. This process starts by receiving packets on the BLE side in the observe mode, and then sends it as a UART message to the WiFi side via the serial input/output port. The UART messages received via the wireless module are transferred to the server on the Internet using the web service method.

The ESP8266 and CC2541 chipsets used in the proposed BLE-WiFi gateway device are cost effective and easily available on the market. If the proposed system had not been chosen, either a high-cost gateway device would be purchased or a BLE-enabled smartphone or tablet would be used to collect the data from the sensor nodes, which has been traditionally used in the market. Although it is possible to use a BLE-supported smartphone or tablet, some difficulties in retrieving the data appropriately or sending it to a server on the Internet may be encountered. A special application must be developed for the smartphone or tablet, and necessary precautions should be considered for the Internet access. If a sensor or sensor group is used, there may be a coverage problem. However, this preferred solution should not be institutional. Consequently, the primary use of the tablet or mobile phone is not an IoT gateway device and will deviate from its intended use.

In this proposed architecture, there is no additional processor or smartphone/tablet because of the single device configuration usage. Therefore, it was possible to transfer BLE advertising packets to the Internet via WiFi at a lower cost. In addition, BLE with extremely low energy consumption is used. Therefore, the system has considerable mobility, a small size, and low energy consumption. The size and energy consumption of the device allows it to be used in different locations and in situations where power supply is offered (existent). Additionally, it is fully compatible with the WiFi network infrastructure. WiFi is a dominant technology in wireless networks, and there are many WiFi-enabled devices in the market. Therefore, it is extremely important to provide compatibility with WiFi networks.

Packet size should be taken into consideration, as it directly affects the overall performance. It is also important to consider the packet transmission frequency and number of peripherals, as well as the wireless media connection capacities and the media in which the data is stored.

This system is vulnerable in terms of security. Not only is it possible for the signals sent from a sensor or a group of sensors in broadcast mode to be received by the BLE-WiFi gateway device, but also by other devices in observation mode in the same environment. This is a disadvantage in the situations requiring security and confidentiality. In order to overcome this security vulnerability, the work will be revised with end-to-end encryption in future versions.

REFERENCES

1. Bello O., Zeadally S. and Badra M. Network layer inter-operation of Device-to-Device communication technologies in Internet of Things (IoT). Ad Hoc Networks, 57, 2017, 52–62.

2. Boualouache A., Nouali O., Moussaoui S. and Derder A. A BLE-based data collection system for IoT. 2015 First International Conference on New Technologies of Information and Communication (Ntic), Mila, Algeria 2015.

3. Chun K.W., Chen Z.H. and Sun C.C. IoT Solution for Public Bicycle System. 2016 IEEE International Conference on Consumer Electronics-Taiwan (Icce-Tw), 2016, 47–48.

4. Collotta M. and Pau G. Bluetooth for Internet of Things: A fuzzy approach to improve power management in smart homes. Computers & Electrical Engineering, 44, 2015, 137–152.

5. Corbellini G., Mangold S. and Schmid S. Two-way communication protocol using bluetooth low energy advertisement frames. SmartObjects - Proceedings of the 1st International Workshop on Experiences with the Design and Implementation of Smart Objects, co-located with MobiCom 2015, Paris, France, 2015, 19-24.

6. De Maso-Gentile G., Baca A., Ambrosini L., Orcioni S. and Conti M. Design of CAN to Bluetooth gateway for a Battery Management System. 2015 12th International Workshop on Intelligent Solutions in Embedded Systems (Wises), Ancona, Italy 2015, 171–175.

7. Gentili M., Sannino R. and Petracca M. BlueVoice: Voice communications over Bluetooth Low Energy in the Internet of Things scenario. Computer Communications, 2016. 89-90, 2016, 51–59.

8. Gubbi J., Buyya R., Marusic S. and Palaniswami M. Internet of Things (IoT): A vision, architectural

elements, and future directions. Future Generation Computer Systems-the International Journal of Escience, 29(7), 2013, 1645–1660.

9. Hawelikar M. and Tamhankar S. A design of Linux based ZigBee and Bluetooth Low Energy wireless gateway for remote parameter monitoring. 2015 International Conferenced on Circuits, Power and Computing Technologies (Iccpct-2015), Nagercoil, India 2015.

10. Komai K., Fujimoto M., Arakawa Y., Suwa H., Kashimoto Y. and Yasumoto K. Elderly Person Monitoring in Day Care Center using Bluetooth Low Energy. 2016 10th International Symposium on Medical Information and Communication Technology (Ismict), Worcester-MAUSA, 2016.

11. Laine T.H., Lee C. and Suk H. Mobile gateway for ubiquitous health care system using ZigBee and Bluetooth. 2014 Eighth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), Birmingham, UK 2014, 139–145.

12. Pyles A., Nguyen D. T., Qi X. and Zhou G. Bluesaver: A Multi-PHY Approach to Smartphone Energy Savings. IEEE Transactions on Wireless Communications, 14(6), 2015, 3367–3377.

13. Siekkinen, M., Hiienkari M., Nurminen J. K. and Nieminen J. How Low Energy is Bluetooth Low Energy? Comparative Measurements with ZigBee/802.15.4. 2012 Ieee Wireless Communications and Networking Conference Workshops (Wcncw), Paris, France 2012, 232–237.

14. SIG B. Developers & Product Manufacturers. 2017. Available https://www.bluetooth.com/develop-with-bluetooth/service-providers/developer-partners (access : November 2018).

15. Espressif Systems. Espressif Espex:a Beginner's Guide. https://tr.scribd.com/document/356929985/0B-Beginner-Guide-to-ESP8266-V0-4-1.. 2014. (access : September 2018).

16. Taşkın D., Taşkın C. and Çetintav I. BLE Single Device Configuration Usage: An Internet of Things Sensor Node. Materials, Methods & Technologies, Volume 11, 2017, 465–473.

17. Tetzlaff, T, Boor M., Witkowski U. and Zandian R. Low Power Network Node for Ambient Monitoring and Heart Rate Measurement. 2014 6th European Embedded Design in Education and Research Conference (Ederc), Milano, Italy 2014, 75–79.

18. Texas Instruments. CC2540 and CC2541 Bluetooth low energy Software Developers Reference Guide. 2015, http://www.ti.com/lit/ug/swru271g/swru271g.pdf. (access : November 2017).

19. Townsend K., Cufi C. and Davidson A & R, Getting Started with Bluetooth Low Energy: Tools and Techniques for Low-Power Networking. Vol. First edition. Sebastopol, CA: O'Reilly Media. 2014.

20. Wang X., Li J.L., Song J., Wang Y.S. and Chen Z.H. Human Physiological Parameter Detection System Based on Android and Arduino. 2016 International Conference on Mechanics Design, Manufacturing and Automation (Mdm 2016), 2016, 535–540.

21. Wang Y.S., Doleschel S., Wunderlich R. and Heinen S. Evaluation of Digital Compressed Sensing for Real-Time Wireless ECG System with Bluetooth low Energy. Journal of Medical Systems, 40(7), 2016.

22. Zachariah T., Klugman N., Campbell B., Adkins J., Jackson N. and Dutta P. The internet of things has a gateway problem. in Proceedings of the 16th International Workshop on Mobile Computing Systems and Applications. ACM, Santa Fe-New Mexico, USA 2015.