ISMSIT2017

1st International Symposium on Multidisciplinary Studies and Innovative Technologies Proceedings Book
(November 2-4, 2017, Tokat, Turkey)
Symposium homepage: www.ismsitconf.org

# Design and Implementation of a Wireless Sensor Node for Real-time Monitoring Applications

Selçuk Ökdem[1*] and Omar Ibrahim Al-Azzawi[2]

*[1]Computer Engineering Department, Engineering Faculty, Erciyes University, Turkey*
*[2]Computer Engineering Department, Graduate School of Natural and Applied Science, Erciyes University, Turkey*
*[*](okdem@erciyes.edu.tr)*

***Abstract –*** Wireless Sensor Networks have many application fields including both industrial and military usages. They are used for monitoring temperature, distance, light, and motion in laboratories and/or application specific environments. This paper focuses on developing a sensor node which provides an easy-to-use way and an economical design. These nodes can be gathered to combine a wireless sensor network. The network can be controlled by the base station and instructed on where to go and what data to collect. This network utilizes the AVR microprocessor based board to store data received by each node and the associated sensors. ZigBee protocol is used for wireless communications. This protocol presents a cheap and small design consuming minimal power, and highly reliable way of data transfer. Data is transferred between each sensor nodet and the base station. A computer is also used to implement instructions and analyze data. The proposed design is described and illustrated in the paper, and performance metrics are presented.

*Keywords – Wireless sensor networks, wireless node, zigbee, arduino, wireless communication*

## I. Introduction

Wireless Sensor Networks, also known as WSNs, have many sensor nodes that interact with the environment. They can sense and analyze many physical characteristics, including: temperature, heat, light, pressure, and motion. These nodes can send and receive information to base station (BS). This base station connects to other networks for storing and analyzing data. There are many functional applications for WSNs, including: home security, smart homes, military, and medicine. Each sensor node along the WSN has two components: a transceiver and a power supply.

A Wireless Sensor Network (WSN) has three advantages: the use of low power, relatively low cost, and small size [1]. These nodes are immediately interactive. They sense, monitor and begin a cooperative process of data transfer through the main sink [2]. This monitoring can be: heat, light, humidity, motion, pressure, sound, or vibration. BS or sink, is an interface between users and the network [3]. This research focuses on designing and implementing a WSN node to monitor temperature, distance, light, and motion in a laboratory. The network is comprised of sensors that are controlled by the BS and instructed the sensors on where to go and what data to collect. The test network uses the Arduino microcontroller board (using AVR microprocessor) to store data received by each node and the associated sensors.

For wireless communication, ZigBee protocol (providing IEEE 802.15.4 protocol specifications) is used. These ZigBee modules are cheap, small, and consume minimal power, which are highly reliable for data transfer. Data is transferred between each sensor node and the base station. A computer is also used to implement instructions and analyze data.

An algorithm was designed for the testing process to manage the transferred data along the WSN for each intended task. This algorithm is presented throughout the test so that each task is performed as it should be and with high efficiency. The AVR microprocessor and the ZigBee module were programmed using a personal computer with specialized programs that are suitable to the proposed network. The paper follows as: describing related works and contribution and main differences to the previous works in the next section, illustrating the proposed design in third section, and finally discussing and summarizing the paper in the following sections.

## II. Related Works

This research was assisted by the work of various researchers, each of which are listed below. The papers studied for this project relate to the design, implementation, and real-time monitoring of the ZigBee technology used in Wireless Sensor Networks.

A team led by Chiara [4] worked in an environmental monitoring technology (using IEEE 802.15.4) in 2009. Their work focused on applications of the technology, as well as design strategies. They also presented the results of a real-time application. Their findings illustrate a wide range of possible applications of WSNs using the “802.15.4” network. Finally, the researchers demonstrated possible sizes for the Wireless Sensor Networks and the possibility for high data transfer.

A team led by Vongsagon [5] provided their findings in 2010 related to developing a standard wireless technology prototype system. They examined a crucial aspect of the tech: monitoring temperature. From a user interface, the team read temperature from every room of a building, including the residents. On their network, the team polled each temperature control from a data coordinator. They placed a wireless sensor known as the Arduino Microcontroller Board on each XBee wireless communication node unit. The team also ran a simple Web data server using an Ethernet interface.

A researcher named Ramanathan and his team [6] used Zigbee in a Wireless Sensor in 2011. They used the system to monitor the physiological condition of a medical patient. The team used a sensor that sent a signal on the WSN using ZigBee and observed any physiological changes. The collected data was then transferred to a computer. When the sensor noticed differences compared to the standard value, the system sent a notification to the medical professionals. This study shows the power of ZigBee as it relates to the health field.

A WSN system that included a monitor, controller, and sensor was crafted for certain specific applications in a 2012 study conducted by Yi-Jen and his colleagues [7]. Their tests demonstrated the various benefits of the ZigBee system, such as its low cost and minimal power consumption. The team also conducted experiments that tested various design elements possible with the system, including a motor control and LCD monitor. The research concluded that the WSN is highly beneficial for data transfer.

The benefits of the ZigBee technology were demonstrated for monitoring temperature control at remote sites by Anas and his team of researchers [8] in 2015. Design elements in the WSN can include temperature-safe covering, added data storage, and perhaps most importantly, the ability to remotely control the environmental temperature using an automatic fan or heater. The team built a PC server and gateway to communicate with the WSN. They tested the system by connecting the gateway to a personal computer and distributed the coordinator and end node within the house. The team received temperature data from various terminals, stored the data in a database, and allowed the system to react independently for temperature changing decisions.

The final study used in this research was conducted by Surabhi and a team or researchers [9] in 2016. This study noted the flow of data through both wireless and transmission as it relates to an agriculture monitoring system. Their study indicated the potential for measuring and reacting to humidity, soil moisture, and external temperature. These findings can benefit farmers financially and also help produce better crops. Table 1 summarizes the main differences of proposed design and other common wireless sensor nodes [10]. The superior benefits are mainly relies on the design of price, on-board sensors, and wide external program memory providing various programming capabilities.

Table 1. Comparison of the proposed design and other nodes

| **Properties** | **Proposed design** | **Telos B Mote [11]** | **MicaZ Mote [12]** | **IRIS Mote [13]** |
|---|---|---|---|---|
| Economy | 77 $ US | 115 $ US | 99 $ US | 99 $ US |
| uP/uC | AT MEGA2560R3 | ATMEGA128 | ATMEGA128 | TI MSP430 |
| Sensors | Motion, Distance, Lighting, Temperature, power level sensor & Can add other | Light, Temperature Pressure | Light, Temperature Pressure, Magnetic | Thermistor, Light Sensor humidity |
| Data memory | 8KB SRAM, 4KB EEPROM | 8KB RAM | 4K RAM | 10K RAM |
| External memory | 256 KB | 128 KB | 128 KB | 48 KB |
| Frequency | 2.4 GHz | 2.4 GHz | 2.4 GHz | 2.4-2.4835 GHz |
| Data-rate | 250 Kbps | 250 Kbps | 250 Kbps | 250 Kbps |
| Range | 120 M | >300M | 75-100 M | 75-100 M |
| Current | 40 mA | 70 mA | 16 mA | 19.7 mA |
| Idle/sleep | -- | -- | 20 uA/1uA | 21 uA/1uA |
| Power | 3.3-5V | 2.7-3.6V | 2.7-5V | 2.7-3.3V |

## III. Developping a Wireless Node

This section presents the configuration and setup of the designed model which utilizes sensors including: motion, distance, lighting, and temperature. This work focuses on the design and implementation of a wireless sensor network and how it has been configured to the designed hardware. The lighting and temperature sensors in the configuration are connected to Arduino module through the analog port pins. The motion and distance sensors are connected to the module through the digital ports.

The configuration is as follows:

1. Arduino Mega board 2560,
2. ZigBee (XBee S2),
3. Motion sensors,
4. Distance sensors,
5. Lighting sensors,
6. Temperature sensors.

It is also important that configuration and setup operation software are used for programming all the devices, including:

1. Configuration of the X-CTU program,
2. Configuration of the Arduino program,
3. Configuration of the processions program.

*X-CTU Configuration:*

This is the program definition for how ZigBee works in the implementation design or circuit. There is an internal microcontroller for each XBee radio that executes the firmware. The firmware's functions include: communication, addressing, utility, and security. One can configure the XBee radios from a computer in many ways. In this configuration, Digit's X-CTU program is used with the XBee Explorer USB adapter. The program is designed to set the firmware files as required to have the XBee modules react properly. The major tabs of the X-CUT GUI are illustrated in Figure 3.2 below. The function of each tab is as follows:

A. PC Settings: This tab allows port configuration and the selection of a specific COM port to fit the desired radio setting.

B. Range Test: This tab checks the range tests between two radio nodes.

C. Terminal: This tab is used to enter the radios' firmware with the AT commands mode. It also allows a terminal emulation program to enter the computer's COM port.

D. Modem Configuration: This tab allows the selection of the needed firmware versions. It also enables programming the radios' firmware with a simple and interactive Graphical User Interface (GUI).

To configured XBee modules on the PC, XBee USB adapters are necessary. Spark Fun XBee Explorer was used in this research. The function of these adapters is direct connection to the PC. After the X-CTU software is downloaded and installed, the Windows configuration follows the following procedure:

1. First connect the XBee USB adapter to the PC without the XBee module inside for the PC to detect the hardware. Then, open Device Manager and go to the COM & LPT section. After expanding the window, the numbers of XBee serial ports plugged into the computer are shown. Note: This step is only required the first time one connects.
2. Next, disconnect the adapters and connect the XBee module to one of them. Then, connect the USB adapter to the PC once more and open the X-CTU software, selecting the serial port. Press "Finish" on the menu to ensure that XBee module is recognized by X-CTU. On the drop-down menus, keep the baud rate at 9600, data bits at 8, parity at None, stop bits at 1, and flow control at "None". These configuration options may be changed to match the needed settings. After the correct boxes are checked, click finish to shift to the Modem Configuration tab.
3. Click Read atop the modem configuration page to load details on the current XBee module configuration.
4. Under Radio Modules, select ZigBee Coordinator AT (not APT) mode.. Next, click on Show Defaults and click on Read once more. This is only done to reset the previous changes.
5. Change the PAN ID from 0 to any hexadecimal value (maybe 1234) for all XBee's in the network. After, click on Write to write the selected configuration settings. Wait for the message in the X-CTU notification area that says the configuration is uploaded to the module without error. Then, open the Terminal tab and leave this window running with the XBee plugged into the PC.

The coordinator device (base station) hardware is shown in Figure 1 below. It consists of the ZigBee (XBee S2) and a ZigBee adapter. These nodes collect all data from the end node devices and display them on the computer. The coordinator node connects to a computer via the USB port.

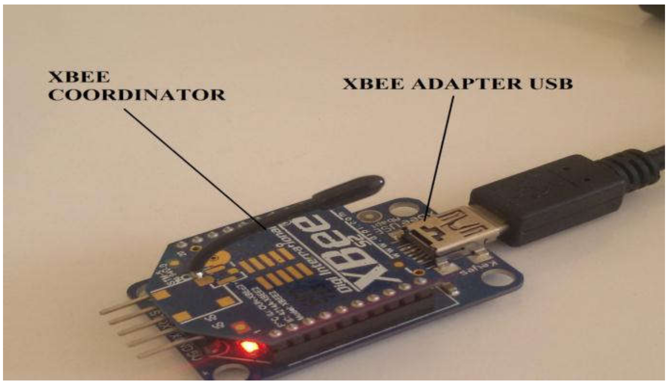


Figure 1. Coordinator Device

1. Connect the second XBee to the USB adapter and repeat Step 2 above. Instead of selecting ZigBee Coordinator AT, select the ZigBee Router AT mode under Function Set. It should be noted that only one coordinator controls the network initialization and control process.
2. Change the PAN ID on the coordinator to 1234. Then click on Write to write the selected settings.
3. Next, set the Channel Verification (JV) to 1 after clicking on Default Setup. This ensures that the XBee is on the correct channel which makes for a more reliable connection to the coordinator. Then click Write.

The router device hardware is shown in Figure 2. This node can only use the ZigBee modem with XBee Shield using Arduino.

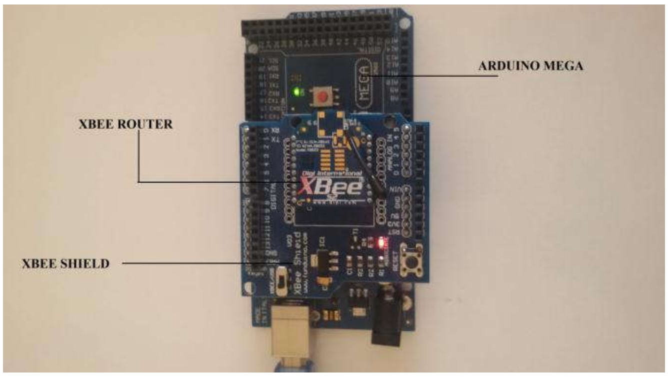


Figure 2. Router Devices with Arduino

Several importation observations on the configuration process should be considered:

- There is only one coordinator on any network.
- The Coordinator can receive data from all routers and end points within the network.
- The routers, including the Coordinator, cannot sleep. End points can sleep, which include the Energy Consumption property.

*Configuring the Arduino Program*:

Arduino is best known for its hardware. However, to use hardware, one must run software. Both the hardware and software share the name Arduino. The combination of the two allow for projects that can sense and control the physical world. The Arduino software is free, open-source, and cross-platform. The hardware boards are cheap, and it's also possible to build your own. The designs for hardware are also open-source. The Arduino software can be downloaded from the manufacturer website of the related Arduino module, as shown in Figure 3.

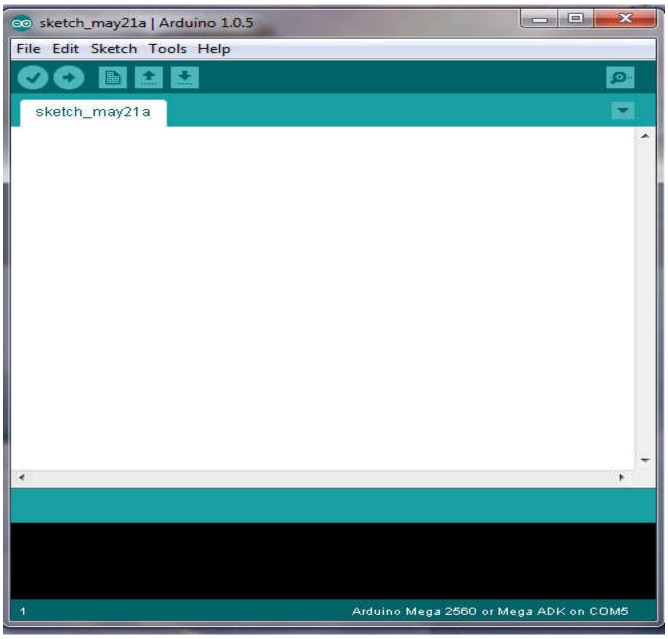

sketch_may21a | Arduino 1.0.5
File Edit Sketch Tools Help
sketch_may21a
1
Arduino Mega 2560 or Mega ADK on COM5

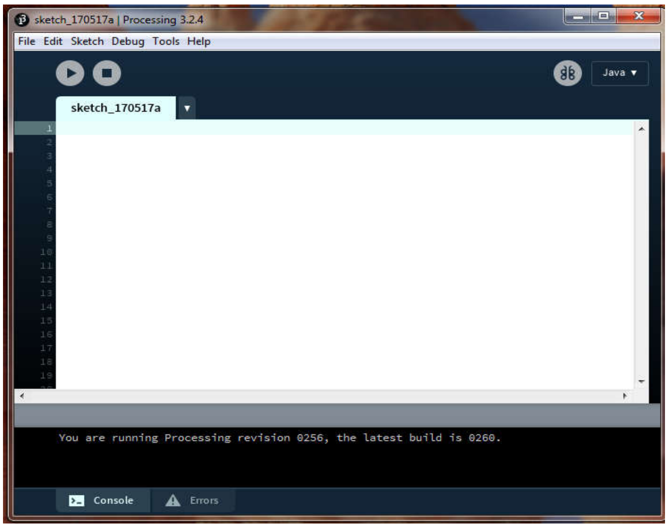
Figure 3. Arduino program

Integrated Development Environment (IDE) Setup:

1. Connect the board to the computer's USB port.
2. Run Arduino IDE.
3. Set the correct port to match your device, and set the Arduino board type to match your device using tools tab (Figure 4).

*Configuration the Processing Program:*

The free software Processing is a Java dialect designed for generating any type of graphics, including visual art and animation. It is available for Windows, Mac OS, and Linux.


sketch_170517a | Processing 3.2.4
File Edit Sketch Debug Tools Help
Java ▾
sketch_170517a
Console
Errors

```
You are running Processing revision 0256, the latest build is 0260.
```

Figure 4. Processing program

The Processing program has its own IDE developed in Java. It is known as PDE (Processing Development Environment).

This program allows you to design a screen or interface in the best format to suit the user application, including customizable color and size. There are many advantages of this software, including the ability to show critical situations through numbers or signs, send this information through the Serial Port, and display on the computer screen.

The software is easy to learn and use, as shown in Figure 4. It is also easy to connect Arduino through handshake and allows for communication between the two software. Data sent from Arduino to Processing through the Serial Port and data received is possible through the ZigBee protocol. The sketches (drivers) for the sensors including motion, temperature, humidity, distance, light, pressure, battery level and other desired ones can be found and downloaded from Arduino website. These sensors can be easily integrated to the proposed node. The output of the sensors is given in the Figure 5.

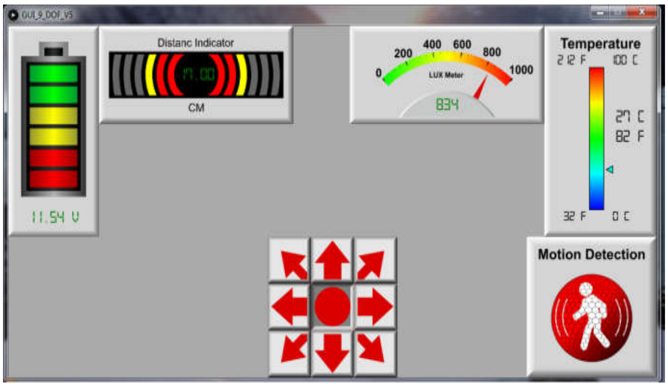


Figure 5. Output sensors device display screen

## IV. DISCUSSION

The print-outs of the measured signals are given by Figure 6 through Figure 9.

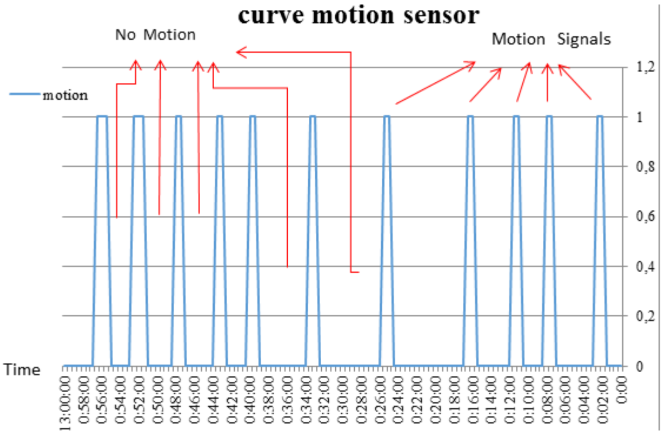


Figure 6. Print-out of motion signal

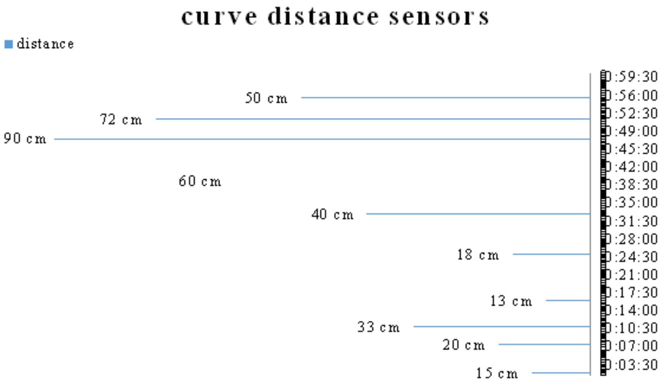


Figure 7. Print-out of distance signal

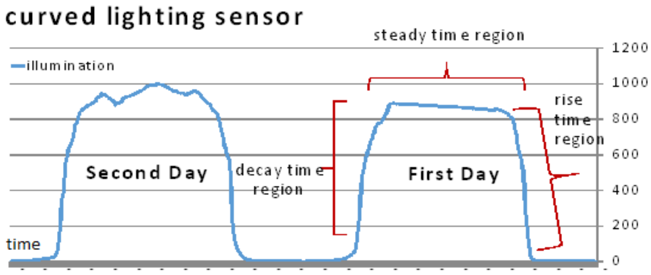


Figure 8. Print-out of light signal

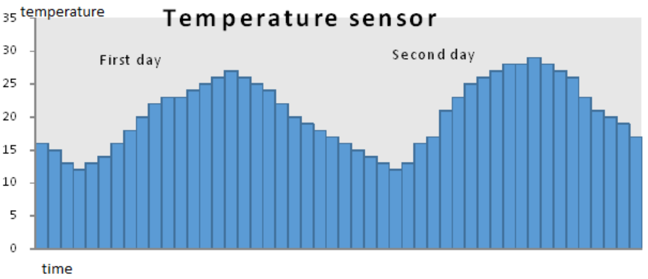


Figure 9. Print-out of temperature signal

As seen from the figures above, observing the outputs of the signals are very easy and flexible to integrate any other tool.

## V. CONCLUSION

Wireless Sensor Networks (WSNs) have a multitude of purposes. This paper set out to illustrate several of these purposes as they relate to reading, storing, and analysing several important physical characteristics. This research used a moving vehicle to read four external characteristics: temperature, distance, light, and motion. Each of these figures was recorded, stored, and transmitted to the base station (BS). This research points to the various ways WSNs can improve working conditions, prevent humans from entering dangerous locations, and read important data that can be used in various fields. Some of these fields include the military, medical industry, for home security, and smart homes. Small vehicles such as the one used in the research above can enter many different areas such as toxic locations, warzones, construction sites, and more, that humans cannot or will not enter themselves.

This research used four sensor nodes attached to an Arduino microcontroller board. There was a special graphical user interface (GUI) designed for reading data, controlling the sensors, moving the car, and setting the parameters for data collection. ZigBee S2 wireless links were used to send data wirelessly throughout the designed network. A personal computer that was suited with a special algorithm for each of the tested parameters was used as the BS.

We noted the differences in external temperature, distance, light exposure, and motion. Each of these parameters was recorded at set intervals, stored in the base station, analysed, and then computed in the form of readable and observable curves. These curves could make data collection easy and allow for the intended purpose of the sensors to be established with ease and along the set goals.

This experiment could be extended in several methods for future work. First, additional sensors could be used to test for different external conditions. Some of these conditions could include the presence of carbon dioxide in the atmosphere, the presence of narcotics, the use of firearms or explosives in a given area, or the number of people present in a certain location. Also, each of the tests could be extended for longer periods of time to predict temperature, motion, or light exposure. It is also possible to make a smaller or larger moving vehicle, or attach the sensors to a different type of vehicle altogether. Each of these extensions of the research would allow for different outcomes, each depending on the purpose of the sensors [14].

## REFERENCES

[1] D. Antolin, N. Medrano, B. Calvo, “A wearable wireless sensor network for indoor smart environment monitoring in safety applications”, *Sensors*, vol. 17(2), pp. 365-378. 2017.

[2] S.N. Pakzad, G.L. Fenves, S. Kim, D.E., “Design and implementation of scalable wireless sensor network for structural monitoring”, *Journal of Infrastructure Systems,* vol. 14(1), pp. 89-101, 2008.

[3] M.A. Matin, “Overview of wireless sensor network”, *Wireless Sensor Networks-Technology and Protocols (Eds: M. A. Matin, M. M. Islam)*. Intech Open, pp. 3-24, 2012.

[4] C. Buratti, A. Conti, D. Dardari, R. Verdone, “An overview on wireless sensor networks technology and evolution”, *Sensors*, vol. 9(9), pp. 6869-6896, 2009.

[5] V. Boonsawat, J. Ekchamanonta, K. Bumrungkhet, S. Kittipiyakul, “Xbee wireless sensor networks for temperature monitoring”, *The Second Conference On Application Research and Development(ECTI-CARD)*, Institute of Technology, Thammasat University, Pathum-Thani, Thailand, pp. 1-6, 2010.

[6] P, R., P, P. M., “Wireless sensor network for continuous a patient's physiological conditions using zigbee”. *Computer and Information Science*, vol. 4(5), pp. 104-110, 2011.

[7] Y.J. Mon, C.M. Lin, I.J. Rudas, “Wireless Sensor Network (WSN) control for indoor temperature monitoring”, *Acta Polytechnica Hungarica*, vol. 9(6), pp.17-28, 2012.

[8] A. Hussien, M.J. Eidan, “Automatic zigbee-based wireless sensor network for real time temperature control”, *International Journal of Wireless Communications and Networking Technologies*, vol. 4(4), pp. 63-67, 2015.

[9] S.N. Singh, S. Kumar, “Automated agriculture monitoring using zigbee in wireless sensor network-A review”, *International Journal of Current Engineering and Technology*, vol. 6(1), pp. 213-215, 2016.

[10] M. Maurya, S. Shukla, R.N., “Current wireless sensor node(Motes):performance metrics and constraints”, *International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE)*, vol. 2(1), pp. 45-48, 2013.

[11] Telos B. (2017), Company homepage on telos mote. [Online]. Available: http://www.willow.co.uk/html/telosb_mote_platform.php

[12] Mica Z node. (2017), Company homepage on mica Z nottelos mote. [Online]. Available: http://tinyos.stanford.edu/tinyos-wiki/index.php/MICAz

[13] Iris mote. (2017), Company homepage on iris mote. [Online]. Available: http://www.rfwirelesssensors.com/tag/iris-mote/

[14] O. Al-azzawi, “Design and Implementation of a Wireless Sensor Node for Real-time Monitoring Applications,” M. Eng. thesis, Erciyes University, Graduate School Of Natural And Applied Science, Department Of Computer Engineering Indian Institute of Science, Kayseri, Turkey, 2017.