Minimum Spanning Tree Topology in Real Zigbee-Arduino Sensor Network

Mourad Ouadou[1], Ouadoudi Zytoune[1], Yassin El Hillali[2], Atika Menhaj-Rivenq[2], and Driss Aboutajdine[1]

[1] LRIT, Associated Unit to CNRST (URAC 29), Faculty of Science, MohammedV University, Rabat, Morocco
[2] IEMN-DOAE, University of Valenciennes, Campus Mont Houy, Aulnoy-les-Valenciennes, France

Abstract. In this paper, we present a real indoor application for improved "Cluster Tree" network using Zigbee sensor nodes integrated with "Arduino" microcontrollers where the main goal is to study the efficiency of our improved clustering topology based on Minimum Spanning Tree (MST) construction under real conditions. Our system is based on many sensor nodes deployed all over a building to monitor temperature and humidity and detect any brutal temperature increase. These nodes are based on simple , cheap and easily programmed equipment. while several simulations using NS-2 and other simulators has been conducted in many other works to study the efficiency of a network based on MST, our application tests this topology in real conditions to ensure it effectiveness in a Zigbee network. Our developed system provides an easy interface for programming and displaying results. All sensed data is sent to the coordinator which forwards it to the user's pc to be viewed by the user. This paper also presents some performances of this discussed topology obtained by real conditions tests.

Keywords: Minimum Spanning Tree, Zigbee, Arduino, Hardware architecture, Temperature-Humidity sensing.

1 Introduction

Nowadays, we assist to great emergence of wireless sensor networks. This type of networks are applied in many fields such industrial and personal like home automation. These networks are self-organizing and consist of a large number of autonomous sensor nodes with low resources that transmit sensed data to the sink or base station [1]. Most of these networks intend to optimize the use of limited energy contained in each node, as well as other tasks like collection, routing and data aggregation to obtain high performances. Among the most emerging application fields for wireless sensor networks we find the indoor sensor networks. In fact, we focus in this work especially on improving the performances of this type of networks aimed for home automation and industrial monitoring by testing a a topology that has demonstrated its effectiveness in simulation in

N. Mitton et al. (Eds.): AdHocNets 2015, LNICST 155, pp. 57–68, 2015.
DOI: 10.1007/978-3-319-25067-0_5

real indoor conditions. For this type of applications, many previous works have demonstrated that conventional wireless standards such as Bluetooth or WLAN [2,3] are not suitable to be used in this type of networks. The first standard taking into consideration the new constraints of the indoor sensor networks, is the 802.15.4 standard. Other reason why we selected ZigBee is because it has been shown that ZigBee network noise is the least compared with other wireless networks in term of SNR [4,5] us referred in figure 1. In fact, in an indoor environment, noise has a strong presence especially in industrial buildings where machines cause high levels of it.

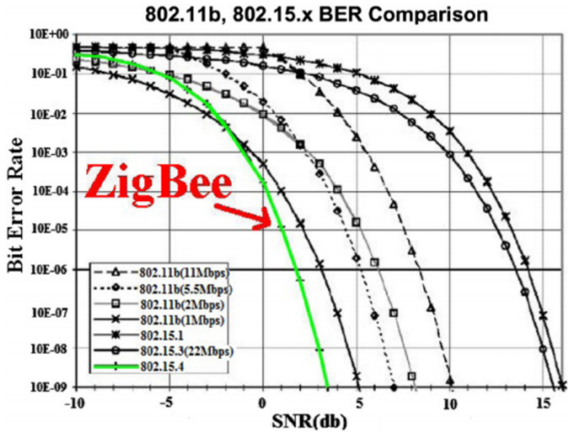

Fig. 1. 802.15.4 performance based on SNR.

Several works has been conducted in improving routing,topology control and data aggregation using simulation tools such as ns-2 [6,7], aimed to improve the efficiency of sensor networks. However in this paper, we focus more on the hardware aspect by presenting a real architecture of the network which allowed us to obtain satisfying results on two aspects: Energy saving and Reliability of data processing. In fact a combination between a Zigbee sensor node and the Arduino platform has allowed a very good reliability of the network as shown in the experimental results. This paper is organized us follows, we first present some concepts in a preliminary notes chapter, then we continue with the second chapter where we present our hardware architecture and system overview. In the third chapter we talk about network organization and communication before presenting in the last chapter our experimental results. At the end of paper we conclude and present our perspectives for future works.

2 Preliminary Notes

2.1 Arduino Open Source

The Arduino platform consists on an open source hardware and software. It is an open-source physical computing platform based on a simple microcontroller board, and a development environment for writing software for the board [8]. Arduino module is a printed board, generally built around an Atmel AVR microcontroller and additional components that facilitate programming and interfacing with other circuits. Each module has at least one linear regulator 5 V and a 16 MHz crystal oscillator. Some of the reason for which we choose the Arduino platform is that Arduino boards are relatively inexpensive compared to other microcontroller platforms and the programming environment is easy-to-use. For our sensor nodes, we used "Arduino Due" microcontroller board based on the Atmel SAM3X8E ARM Cortex-M3. It is the first Arduino board based on a 32-bit ARM core microcontroller. It has 54 digital input/output, 12 analog inputs, 4 UARTs, a 84 MHz clock, an USB OTG capable connection, 2 DAC, 2 TWI, an SPI header, a JTAG header, a reset button and erase buttons.

2.2 Minimum Spanning Tree

A minimum spanning tree (MST) of an undirected graph is a graph that spans all the nodes as vertices and contains no cycles [9]. In application for the sensor networks, several algorithms for routing and topology control based on MST have been developed. This algorithms use few methods that allow the construction of MST based topology as "Relax" algorithm, "Adjust weight" algorithm, "Adjust tree" algorithm, "STP tree based" algorithm and others. In general, spanning tree algorithms have shown to be energy efficient for different extreme sensor network conditions. For our network, we used an improved "Adjust Tree" algorithm for MST topology building for our network. This algorithm calculates the cost between two relations and continues until all the possible pairs have calculated there costs. It sorts all edges in the increasing order, and considers all these edges for inclusion in the tree. It initially has all vertices but no edges. An edge is included into the tree if and only if its addition does not create a cycle in the already constructed tree.

2.3 Temperature and Humidity Sensing

In many indoor systems, especially in industrial field, the temperature-humidity control is crucial. In fact, it is essential in the industry to control those parameters to ensure the safety of equipment and staff. This type of sensors provides temperature information to the controller so it can make some urgent decisions like over-temperature shutdown. it exists a lot of kinds of passive and active temperature sensors that can be used to measure system temperature. We used the "Aosong dht22" temperature-humidity based on Polymer capacitor. It uses exclusive digital-signal-collecting-technique and humidity sensing technology, assuring its reliability and stability. Its sensing elements are connected with 8-bit single-chip computer [10].

3 System and Hardware Overview

3.1 System Overview

The system we present in this paper is an energy independent indoor sensor network designed for an automation system in order to monitor temperature and humidity inside buildings. In this system, sensors needs a battery and the communication protocols have to be energy efficient. This system uses 802.15.4 standard to perform communications between the different components of this network. Our system is based as shown in figure 2 by a set of sensor nodes. These nodes are divided into three main categories:

Sensing Unit. This unit is composed by a big number of sensor nodes. This unit provides the task of sensing and routing. These nodes are composed into two kinds, routers and end devices. As presented in 802.15.4 standard [11], the "end devices" are only programmed to sense and send data to the next hope toward control unit, the "routers" are programmed to receive data and decide which node it has to be sent for the next hop toward the control unit.

Control Unit. This unit is composed by a coordinator node connected to user's pc. The coordinator has the main task of organizing the network and communications. It has the capability to add or remove any node from the Zigbee network, allow address to the nodes and create and control the topology. From the user's pc, we can make some occasional modifications wich are transmitted to the coordinator. This modifications can be in some cases a network re-building using a different topology.

3.2 Hardware Architecture

There is a large amount of works on developing sensor nodes and new communication devices. Many sensor nodes were developed by the researchers [12,13,14], and others are commercial.

Sensor node architecture depends on many components that goes into the constitution of the node. In table 1, we present a comparison between the most used Zigbee sensor nodes. The two main components in a Zigbee sensor node are the microcontroller and the RF module. The microcontroller defines the programming environment, meanwhile the choice of RF module has a major

Table 1. Comparison between some Zigbee sensor nodes

Commercial nodes	Microcontroller	RF Module	Comsumption	Indoor Range
Texas Instruments	CC2530	SmartRF05EB	24-29 mA	108 ft
Microchip	PIC24	MRF24J40	19-23 mA	110 ft
Digi	Arduino	Xbee	40-50 mA	100 ft
MEMSIC	ATmega128L	Micaz	11-19.7 mA	82 ft

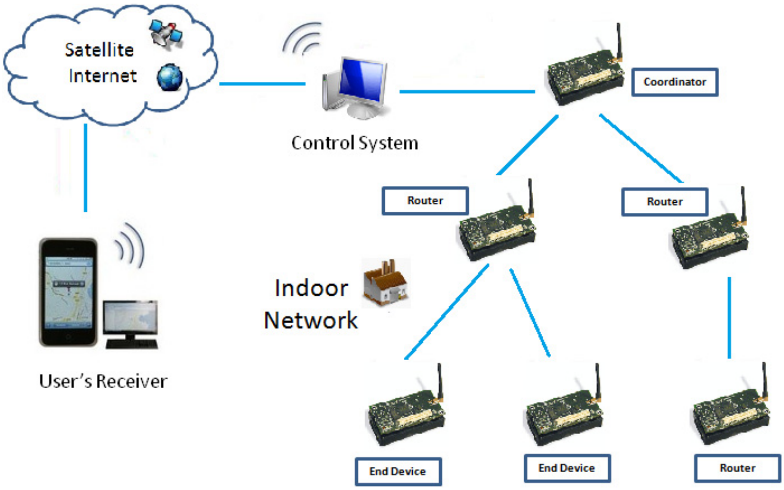

Fig. 2. System overview.

influence on the energy consumption and the organization of network since it defines the distance range for transmitting for indoor and outdoor areas. Our sensor network is composed by 13 sensor nodes composed each one by a circuit board attached to many components.

The principal component attached to our circuit is the MRF24J40MA showed in figure 3. All sensor nodes, except the coordinator uses this RF transceiver. The coordinator uses the MRF24J40MD which has a higher range. In fact, according to our tests, the range of the MRF24J40MD can reach 250 feet in a indoor area. This distance is more than enough for the coordinator in a building to be located a little far from the rest of the network, near to the user's pc.

We also used the "Aodong Dht22" temperature-humidity sensor based on Polymer capacitor. It uses exclusive digital-signal-collecting-technique and humidity sensing technology, assuring its reliability and stability. All nodes, including the coordinator are equipped by those sensors. The node includes also a 8-bit switch. The switch enabled us to allow a mac address to the different nodes manually. To build this nodes, we designed a circuit board wish design is shown in figure 3 in order to connect all components to the "Arduino Due" card.

This circuit board that includes all the components listed above is integrated to the Arduino circuit. We show in figure 4 one of the sensor modules that includes other electronic components: An 9A103G network resistor for the switch, one 10K resistor for the sensor, 2 capacitors and one reset button.

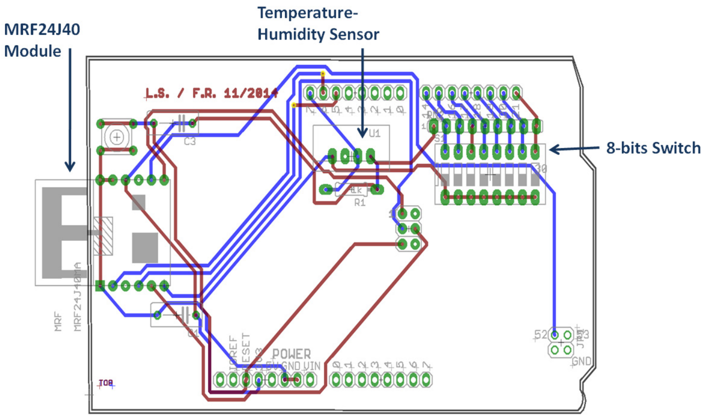

Fig. 3. Circuit board for sensor node

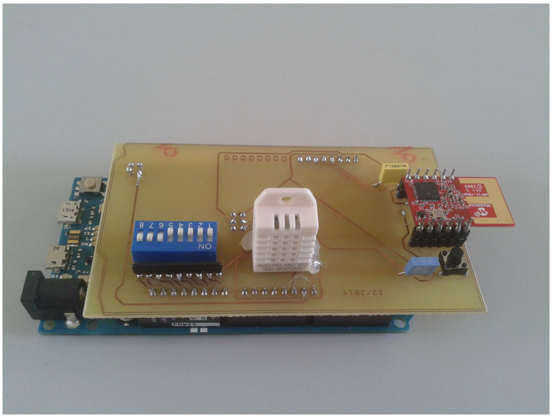

Fig. 4. MRF24j40Ma/Arduino sensor node

4 Software and Network Presentation

4.1 Software Interface

As seen in previous chapter, the sensor node's architecture in our system is based on Arduino microcontrollers, thus we used Arduino software for programming the MRF24J40 sensor nodes. In 802.15.4 standard, two types of zigbee devices are defined: Full-function device (FFD) and Reduced-function device (RFD). Note that, as explained in the next chapter, our topology is based on Cluster-tree. So we used 2 Arduino program files: Coordinator.ino, Non-Coordianator.ino. In the figure 5 we present the Arduino programming interface.

Fig. 5. Arduino programming environment

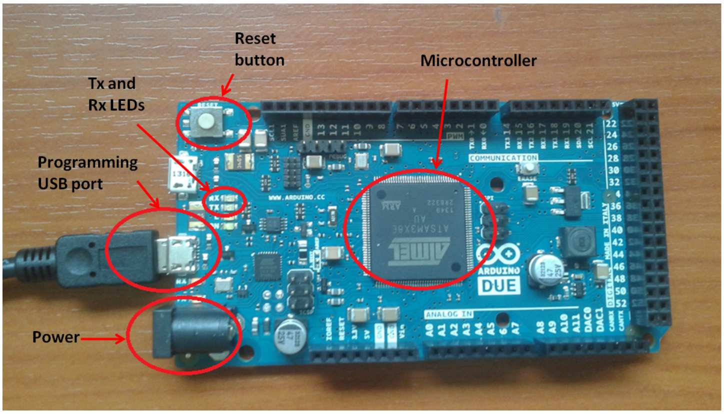

Fig. 6. "Arduino Due" principal components for programming

First, the code has to be uploaded to Arduino card. For uploading the code we use the programming port in figure 6. The Coordinator code has to be uploaded in the Arduino card that will be connected to the MRF24j40MD RF module. The non-coordinator code has to be uploaded to the other circuit boards that include the MRF24j40MA RF modules. When uploading the code in the Arduino board, the Tx and Rx LED are on. After that, we can disconnect the Arduino card from the PC, attach it to the circuit board including the Zigbee module, and connect the whole node to power input also shown in figure 6.

Once the reset button is pressed, the node self initializes and starts transmitting data according to steps explained in the next chapter.

All data sent by the sensor nodes in the network are received at the coordinator. The coordinator send the received data to the control system machine as shown in the system overview chapter. The control system's interface shows the coordinator's running interface. When the coordinator starts running, it initializes the "PAN ID", defines itself as coordinator using the "RXMCR" register of MRF device, assigns a 16 bits address to itself and starts waiting for other nodes the send "join request". Once a join request is received in the channel, the coordinator checks that the RF module of the node is the MRF24J40MA, and assign an address and the PAN ID to the device, and add it in to the network topology. A "request response" is then sent to the device. Once this steps performed successfully, the coordinator starts receiving data from the added node. This operation is then repeated for all the sensor nodes that request joining the network.

4.2 Network Deployment

In this section, we present how the sensor network is organized. The network topology used is the "Cluster Tree" from 802.15.4 standard. To obtain our topology, we made some modifications on the tree construction algorithm in order to create a minimum spanning tree where the leaves are "end devices", and the other nodes are "routers". The nodes sends a "Hello Packet" to the coordinator including the MAC address obtained from the 8-bits switch and their positions in the building. To create and control the topology, as we mentioned before, the improved adjust algorithm is used to create a minimum spanning tree. we Suppose a graph $G = (V, E)$ representing the network, where vertices V represents sensor nodes and edges E represent the link between two connected nodes [15]. In figure 7, we present the flowchart of the operations at the coordinator and the end devices. Note that the router have a similar functioning except the difference that it can receive packets from children nodes and send it to parent node.

To test the performance of your system, we deployed our network in an indoor environment. Figure 8 shows how network is deployed and topology is constructed inside the building of Faculty of Sciences of Rabat where we realized our network.

5 Experimental Results

As mentioned in sections before, we performed a field test for temperature and humidity sensing using MRF24J40-Arduino sensor network inside the building of Faculty of Sciences of Rabat. Our system was programmed in order to sense temperature and humidity for all nodes every minute. The coordinator sends obtained data from other nodes to user's pc. We show in figure 9 the results obtained in the user's PC that shows all the operations happening at the coordinator.

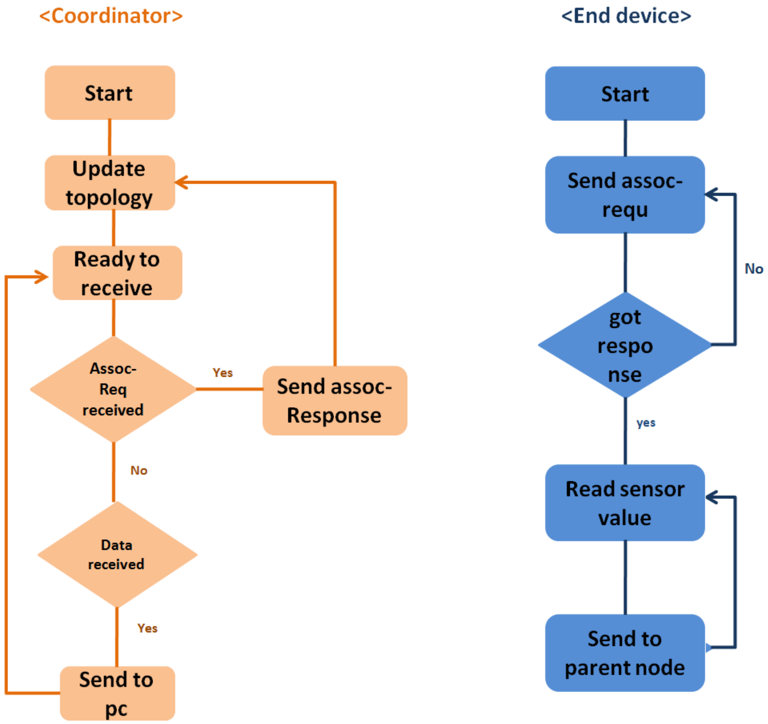

Fig. 7. Coordinator and End Device flowcharts

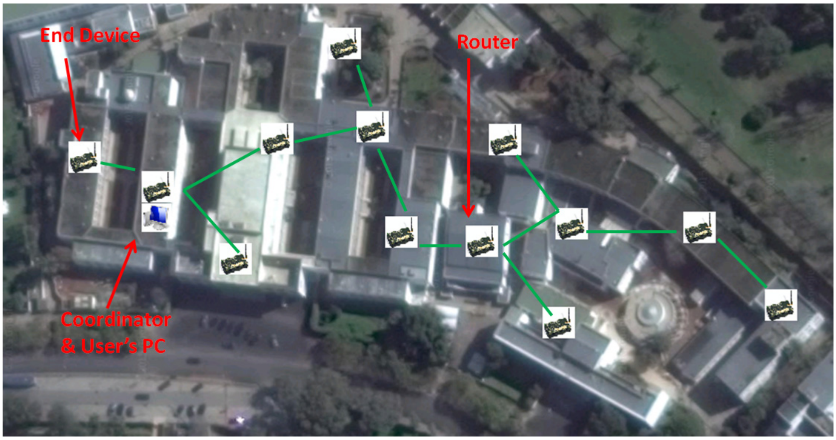

Fig. 8. Sensor network deployment in Faculty of Sciences of Rabat building

For the main performance results, we can view in figure 10 the packet loss ratio in the network according to the number of sensor nodes in the network for

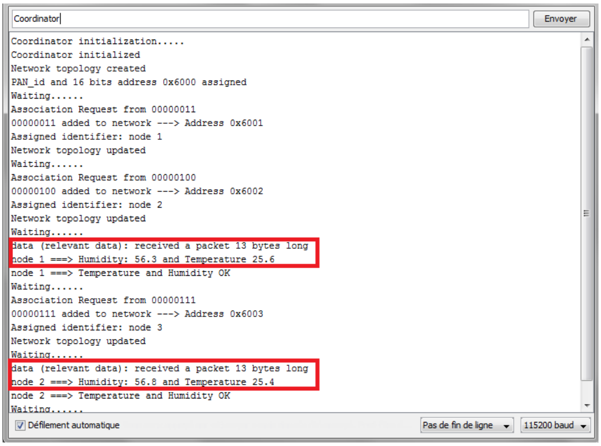

Fig. 9. "Coordinator operations at user's PC interface

typical Cluster Tree and Minimum Spanning Tree Cluster topologies. Our first tests started with 5 nodes and then we started adding nodes to reach 9 and then 13 by testing network performances. Our improved MST topology shows better results in terms of packet delivery ratio than the typical Cluser tree topology.

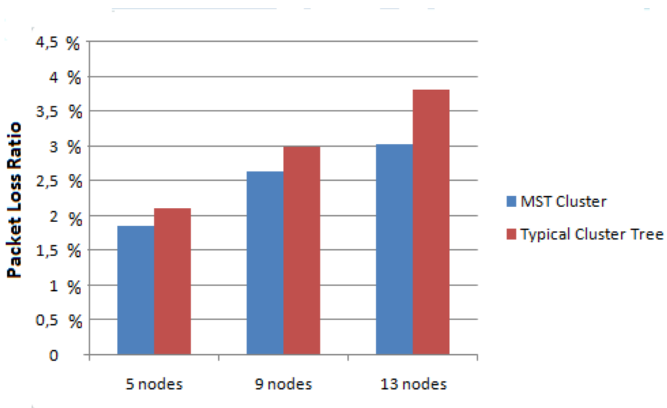

Fig. 10. Packet loss ratio

We also compared the average of power consumption in the network as shown in figure 11 by summing the energy consumption for all sensor nodes. Once again our improved topology shows better results for our network which means longer lifetime duration for our network.

In summary, the results show satisfactory performance of the network using our improved MST topology in real indoor Zigbee conditions in term of function-

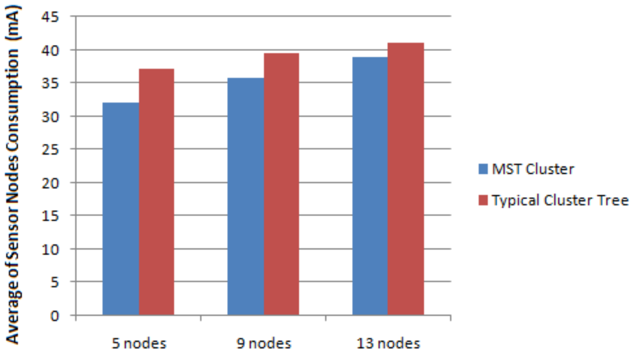

Fig. 11. Average power consumption

ality, packet deliverance and power consumption. In fact, the packets containing sensed data are routed quickly to the coordinator node, wish allows us to follow in real time the evolution of the temperature and humidity inside the building. The interface allows to visualize at the same time the evolution of the temperature in the building and the network status by showing the nodes joining our personal area network. The average of energy consumption, which represents a major issue for energy independent sensor networks, has been reduced by about 11 per cent compared to typical cluster tree topology in our tests, which is considered an important gain in term of network's lifetime.

6 Conclusion and Perspectives

In this paper, we presented how we built a prototype of a zigbee sensor network based based on minimum spanning tree clustering using MRF24J modules and "Arduino microcontrollers" to test performance of this topology. We presented the architecture and components of the nodes and how the easy programmed nodes are deployed to build a efficient sensor network for home automation. We made several tests that showed the reliability of our system based on easy interface for temperature-humidity monitoring. The tests showed also satisfactory results for our system in therm of functionality and network performances as packet delivery ratio and energy consumption. Extensions of our current work includes adding new technologies for energy harvesting in order to prolong network's lifetime.

References

1. Ulema, M.M.: Coll. Wireless sensor networks: architectures, protocols, and management. In: Network Operations and Management Symposium (2004)
2. Bargh, M.S., De Groote, R.: Indoor localization based on response rate of bluetooth inquiries. In: Proceedings of MELT 2008, pp. 49–54 (2008)

3. Gomez, C., Paradells, J.: Wireless home automation networks: a survey of architectures and technologies. IEEE Communications Magazine 48, 92–101 (2010)
4. Shuaib, K., Alnuaimi, M., Boulmalf, M., Jawhar, I., Sallabi, F., Lakas, A.: Performance Evaluation of IEEE 802. Journal of Communications 2 (2007)
5. Alnuaimi, M., Shuaib, K., Jawhar, I.: Performance evaluation of IEEE 802.15.4 physical layer using MatLab/Simulink. In: Innovations in Information Technology (2006)
6. Melodia, T., Pompili, D., Akyildiz, I.F.: On the interdependence of distributed topology control and geographical routing in ad hoc and sensor networks. IEEE Journal on Selected Areas in Communications 23(3), 520–532 (2005)
7. Li, M., Li, Z., Vasilakos, A.V.: A Survey on Topology Control in Wireless Sensor Networks: Taxonomy, Comparative Study, and Open Issues. Proceedings of the IEEE 101(12), 2538–2557 (2013)
8. Schubert, T.W., D'Ausilio, A., Canto, R.: Using Arduino microcontroller boards to measure response latencies. Behavior research methods. Springer (2013)
9. Javier, F., Martnez, O., Stojmenovic, I., Garca- Nocetti, F., Solano-Gonzlez, J.: Finding minimum transmission radii for preserving connectivity and constructing minimal spanning trees in ad hoc and sensor networks. Journal of Parallel and Distrubuted Computing 65, 132–141 (2005)
10. Liu, T.: Digital-output relative humidity and temperature sensor/module DHT22. Aosong Electronics Co.,Ltd.
11. Microchip Technology Inc . MRF24J40 Data Sheet, IEEE 802.15.4, 2.4 GHz RF Transceiver (2010)
12. Otto, C., Milenkovic, A., Sanders, C., Jovanov, E.: System Architecture of a Wireless Body Area Sensor Network for Ubiquitous Health Monitoring (2006)
13. Zulkifli, N.S.A., Harun, F.K.C., Azahar, N.S.: XBee wireless sensor networks for heart rate monitoring in sport training. In: 2012 International Conference on Biomedical Engineering (ICoBE), pp. 441–444 (2012)
14. Boonsawat, V., Ekchamanonta, J., Bumrungkhet, K., Kittipiyakul, S.: Xbee wireless sensor networks for temperature monitoring. In: ECTI-CARD (2010)
15. Khan, M., Pandurangan, G., Kumar, V.S.A.: Distributed Algorithms for Constructing Approximate Minimum Spanning Trees in Wireless Sensor Networks. IEEE Transactions on Parallel and Distributed Systems 20(1), 124–139 (2009)