IPv6 Wireless Sensor Network Gateway Design and End-to-End Performance Analysis

Gopinath Rao Sinniah, Zeldi Suryady,
Reza Khoshdelniat, Usman Sarwar, Mazlan Abbas
Wireless Communication Cluster
MIMOS Berhad
Kuala Lumpur, Malaysia
{*gopinath.rao, zeldi.suryady, reza.khoshdelniat, usman.sarwar, mazlan.abbas*}@*mimos.my*

Sureswaran Ramadass
National Advanced Centre of IPv6 (NAv6)
Universiti Sains Malaysia (USM)
Pulau Pinang, Malaysia
sures@nav6.usm.my

***Abstract*—The need for low power personal area network (LoWPAN) devices to be connected to the Internet is increasing due to the demand and proliferation of new applications. Previously, these devices didn't have the need to be connected to Internet. With the introduction of IPv6 over Low power personal area network (6LoWPAN) and the push for Internet of Things (IoT), these devices are now reachable using the common TCP/IP stack. Gateway is an important component to ensure that the packets from LoWPAN network are properly routed to the Internet. This paper provides a new gateway architecture to support 6LoWPAN stack and the performance analysis for end-to-end communication in an office environment. The architecture can be used for implementation in various 6LoWPAN related applications. Performance is measured on the latency and transmission success rate. The experiment results shows that the communication between WSN and client using the 6LoWPAN gateway is successful. Besides that, the success rate is 100% for 1 hop and slightly lower in 2 hops. The latency rate between 100 and 135 ms is acceptable and comparable with existing prior art which is 125 ms on average.**

Keywords-6LoWPAN; Wireless Sensor Network; Gateway; IPv6; 802.15.4.

I. INTRODUCTION

One of the growing sectors in wireless technology is IEEE802.15.4 Low Power Personal Area Network. IEEE802.15.4 is the basis for the ZigBee [13], WirelessHART [14], ISA100.11a [15] and MiWi [16]. These existing standards were created to provide connectivity in Personal Area Networks (PAN) area without having connectivity to the Internet. This is because small devices with low resources are thought to be incapable to have TCP/IP stack and also because the needs to be connected to the Internet were not matured.

Knowing the fact that existing TCP/IP is too heavy to be used in IEEE802.15.4 devices, 6LoWPAN [1] working grouping was created to provide a solution. The Working Group (WG) stated that the solution would be "pay as you use" header compression method that removes redundant or unnecessary network level information in the header. Some of the information can be derived from link-level IEEE802.15.4 header. Hence the 40 bytes IPv6 header was reduced to 2 bytes. This is achieved by reusing the link layer header information. The reduction of the header size is necessary as the total header size of IEEE802.15.4 is only 127 bytes which is too small to accommodate the entire 40 bytes IPv6 header.

The solution by the 6LoWPAN Working Group does not provide an end-to-end communication between the nodes and external devices in the Internet. It is because the header format of 6LoWPAN is different than the standard IP header. Therefore, a gateway or an intermediary device is required to provide a conversion between the 6LoWPAN and IP Header. The adaptation layer that is positioned between the link and network layer, provides the header compression for 6LoWPAN nodes. A gateway architecture was proposed [2] to provide solution for this. It provides an interface for communication between the IEEE802.15.4 nodes that uses 6LoWPAN stack to the external network that has interfaces such as WiMAX, Ethernet and WiFi. The gateway is configured so that it can process both the data that is sent periodically by the nodes and also request from the client. This paper adds contributions to [2], by providing the detail gateway architecture and performance analysis of the gateway. Figure 1 shows the overall communication scenario and the communication stacks between 6LoWPAN nodes, gateway and end user.

This paper is organized as follows. Section II reviews the existing solutions related to WSN gateways. Section III provides the gateway architecture and the communication between various components. Section IV discusses the implementation for the gateway, while Section V gives the experiments that were conducted and the results. Finally, conclusion and future research work are presented in Section VI.

 ISBN: 978-1-61208-207-3

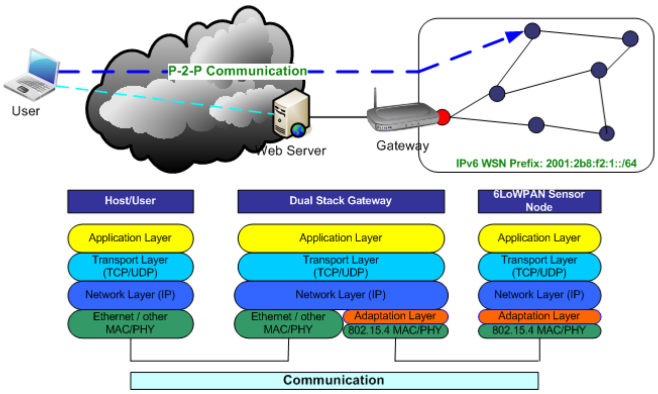

Figure 1. Interconnection between WSN nodes and end user and the communication stacks

II. Existing Solutions

There are several gateway architectures that were proposed for various implementation scenarios. One of the applications is for AC energy usage monitoring using 6LoWPAN. Jiang et al. [4] discussed the application that uses TinyOS and blip. Edge routers were used to route data to a database and uses a web server for visualization of the data. Wenbin et al. [5] developed WSN gateway specifically to monitor forest environment. Information from the sensors is sent to a monitoring centre using GPRS module. Jara et al. [6] introduced a WSN architecture that uses mobile nodes to collect healthcare information. 6LoWPAN gateway was used to connect the nodes to a database. All these solutions have little information on the gateway design and didn't provide the operation of the gateway.

Dun-Fan et al. [3] proposed gateway architecture for environmental monitoring which connects WSN with external network and shares the data collected using web services. The paper didn't explicitly mention the addressing of the nodes and communication between the nodes and external network. It is stated that ZigBee proprietary protocol was used for WSN. This is a drawback as the end users cannot directly communicate with the nodes.

There is an implementation [7] that uses 6LoWPAN short ID which is the MAC address. The internal node retrieves the destination MAC address by querying the gateway using the destination IPv6 address. Gateway retrieves link layer MAC address from the destination address provided by the internal node. This is unnecessary process as the nodes can directly send the data using the destination address and it is not practical as not all global address generated using link layer MAC address. Zimmermann et al. [8] introduces a one-to-one translation between link local address and global address at the gateway. They use a DNS-ALG like server to intercept the DNS query to assign link local address to internal node. If the DNS query could not be intercepted, communication would be disrupted.

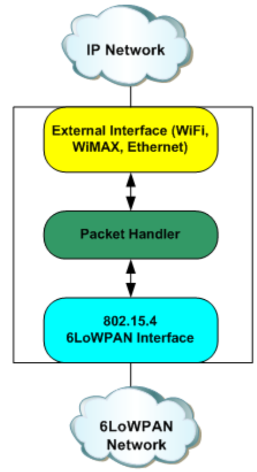

Figure 2. 6LoWPAN Gateway Modules

Jin et al. [12] proposed an interoperable architecture between NEMO and 6LoWPAN focusing on routing scheme. The nodes are configured with global IPv6 address and as such translation of header is not required. In our solution, we propose a solution for nodes that uses MAC address for communication. Besides that, we focus on the performance analysis on real testbed compared to the simulation results by [12].

III. Gateway Architecture

A complete end-to-end architecture could consist of sensor nodes, a gateway, database server or web server and end users. The gateway is designed to support two kinds of standard communications:

- Pull communication method - IPv6 clients request data from sensor node in 6LoWPAN network.
- Push communication method - Sensor nodes periodically send data to an external IPv6 device. The external IPv6 device in this system could be any remote station or server.

Based on dual stack protocol, the gateway is designed to have 3 modules as shown in Figure 2. The PHY/MAC for multiple interfaces that connect to external IP network is defined as external interface module, 6LoWPAN PHY and MAC layer is defined as 6LoWPAN Interface Module and all the services that might be implemented on top of adaptation layer which are network layer, transport layer and application layer reside in Packet Handler Module of the architecture.

The function of the three modules are explained briefly below.

- **6LoWPAN Interface (WSN) Module** - This module consists of IEEE802.15.4 compliance hardware which has the 6LoWPAN stack on it. The module is responsi-

 ISBN: 978-1-61208-207-3

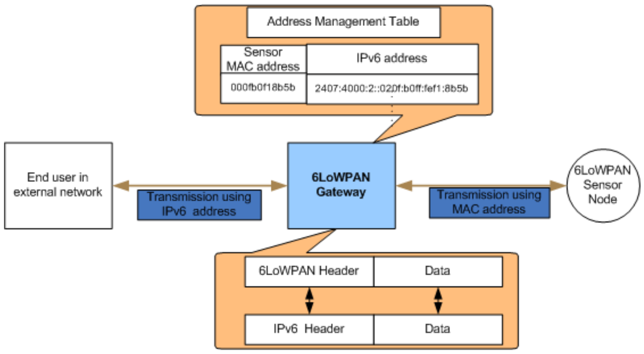

Figure 3. Address management table in the Gateway

ble for handling connectivity and data transmission of 6LoWPAN network using IEEE802.15.4 standard.

- **Interface Module** - This module defines the Physical and MAC layer of any interface that provides connectivity to external IP network. Therefore, the role of this module is to offer functionalities required to ensure connectivity to external public IP network. Some of the interfaces might provide connectivity to LAN/Wireless LAN (e.g., Wi-Fi), while others can provide connectivity to backhaul internet (e.g., Ethernet or WiMAX).
- **Packet Handler Module** - This module provide services to handle both 6LoWPAN and IPv6 packets. This is a significant module that bridges all the interfaces that connects to different networks. Since most of the main processes occur in this module, the service module has major responsibility integrating the 6LoWPAN network with the IP network through other external interfaces. The main purpose of this module is to provide functionalities for handling standard IPv6 packet from external network as well as 6LoWPAN packet. Both IPv6 and 6LoWPAN packets are analysed and processed accordingly. 6LoWPAN packets are transformed to IP packet and vice versa to enable smooth communication between 6LoWPAN nodes and external network.

Packets arrive at the gateway both from external network and 6LoWPAN network, would be first identified based on the address. If the packets are from external network, the gateway would find the destination address in the mapping table. The address would be translated into matching MAC address and the data is copied to 6LoWPAN header and sent to the node. If there is no matching address in the table, the request from the external network would be discarded. The same process is applied when a packet arrives from 6LoWPAN network. The MAC source address is replaced with the IPv6 address of the node and packet is sent using IPv6 header. The translation and mapping table is given in Figure 3.

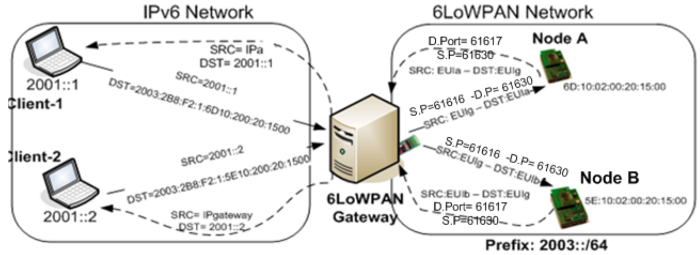

Figure 4. Gateway handling one-to-one communication

A. Gateway Communication

There are two scenarios for pulling sensor data; one client to one sensor node communication and many clients to one sensor node communication. For both the scenarios, a table is created to handle the packets that arrive at the gateway. Gateway maintains the entry in an *Address Information Table*, which will be used to route the sensor nodes' response packet back to the corresponding users. *Address Information Table* consists of ID number of packet, source address (Client's IPv6 address) and destination address (Sensor Node MAC address), port number allocated and the status. The status could be that the packets are already sent to sensor node, gateway already replied to client's request or and packet pending for transmission awaiting reply from sensor node for earlier requested information. Packets that are destined to the same node would be queued and not immediately transmitted to avoid collision. With the use of this table, retransmission of packets would be reduced and this will save energy in the nodes. An example of one-to-one communication is given in Figure 4. Different port numbers are used to differentiate the sensor's traffic from both the schemes. RFC 4944 [11] defines a well-known port range (61616-61631) for UDP packet in 6LoWPAN. In this implementation, the ports used are

- Port 61631 is used at the gateway to receive data from sensor nodes in push based method.
- Port 61616 is used by the gateway to send data to the sensor nodes in pull based mechanism.
- Port 61617 is used by the gateway to receive data from sensor nodes in pull based mechanism.
- Port 61630 is used by the nodes to receive the request from the external node through the gateway.

In the push method, sensor nodes send data to a fix destination port, 61631. All the data arrive at the gateway at that port would be automatically forwarded to a preconfigured destination address of a collector or database server. This is shown in Figure 5.

IV. Gateway Implementation

A testbed was created to validate the gateway architecture and to measure the end-to-end performance as shown in

 ISBN: 978-1-61208-207-3

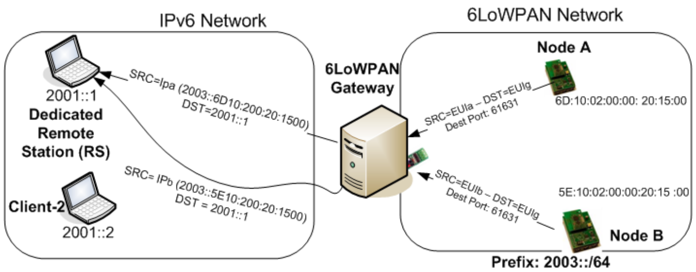

Figure 5. Pushing data from sensor to external node

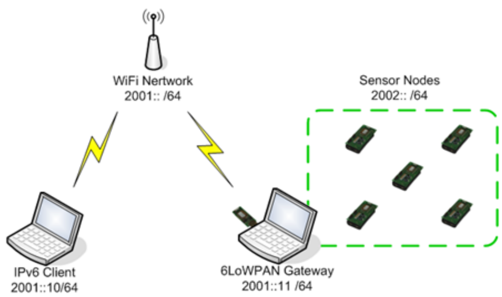

Figure 6. Testbed for validating and performance measurements

Figure 6. The tests were conducted in an indoor lab environment with over 20 active WiFi Access Points operational which is detected using a Network Stumbler software [11]. The sensor nodes that were deployed provide readings for temperature and light intensity measurements.

The setup consists of nano router and sensor nodes developed by Sensinode Inc. [9] as our hardware platform. Gateway is a laptop computer with Linux OS and has three interfaces; a nano router for the wireless sensor network and WiFi and Ethernet interface that connects to the IPv6 network. Nano router is a USB device that is attached to one of the available USB port in the gateway. Packet Handler module explained earlier is configured and executed on the gateway. The sensor nodes are installed with the free real-time operating system (FreeRTOS) with the NanoStack software module which consists of 6LoWPAN stack with added features. Each of the sensor node has 2 AA batteries. The modules were developed using c programming language. The communications for both push based and pull based schemes are maintained through the use of a gateway.

A client laptop was also used to retrieve sensor data to verify the bidirectional communication. To validate the performance, tests with different settings were conducted with different data sizes. Furthermore, to test the bidirectional communication, a ping message was sent from the gateway and using the reply, the latency was calculated. Table 1 provides the properties for the tests.

V. System Performance and Evaluation

As described earlier, the request from a client will be forwarded by the gateway using a simple client as shown

Table I
Performance measurement properties

Properties	Details
Network Size	4-8 nodes for 1 hop away. 2x2, 2x4 and 2x6 for 2 hops
Distance	3 meters for each hop
Data Sampling intervals	20 seconds
Duration	120 samples (1 hour)
Message size	4, 8, 16, 37 bytes
Measurements	Transmission Success Rate and Latency
Method	Start with 1 node and gradually increase the nodes while sending data simultaneously

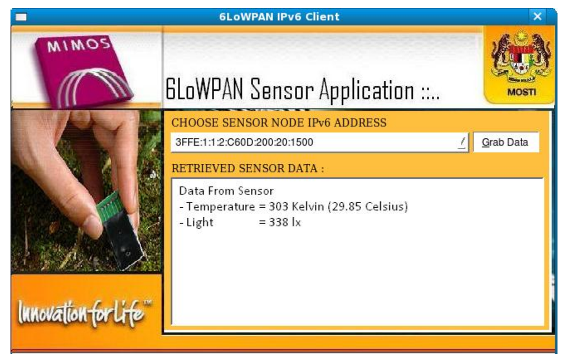

Figure 7. IPv6 client application to read data directly from sensors. ©2009 MIMOS Bhd. All Rights Reserved

in Figure 7 [2]. All the sensor nodes' IPv6 addresses are listed in the client and when a particular IPv6 address is selected, a request is forwarded to the gateway which will then do the necessary actions. The temperature and light reading from the sensor will then displayed on the client. This shows the success of bidirectional communication (Pull based mechanism). In the push based mechanism, the data is periodically sent to a web server and the data is displayed using a web browser as shown in Figure 8.

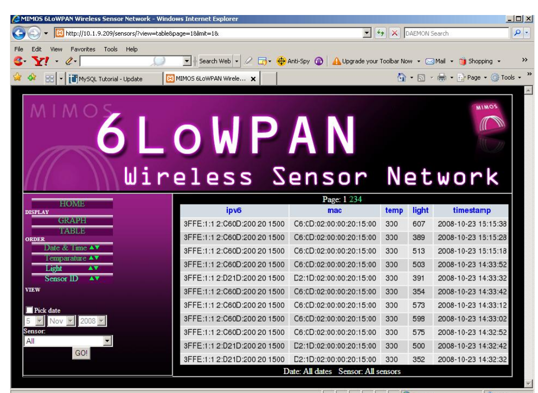

Figure 8. Display sensor information using web browser. ©2009 MIMOS Bhd. All Rights Reserved

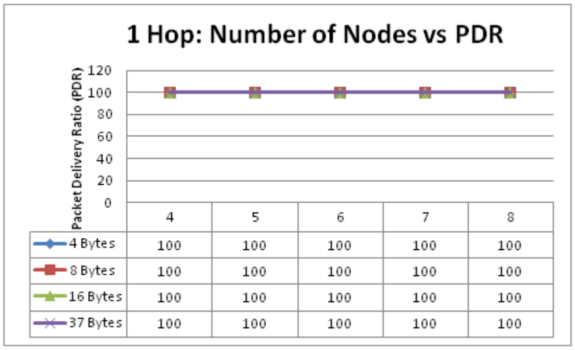

Figure 9. Performance of packet delivery rate against number of nodes and different packet sizes for 1 hop

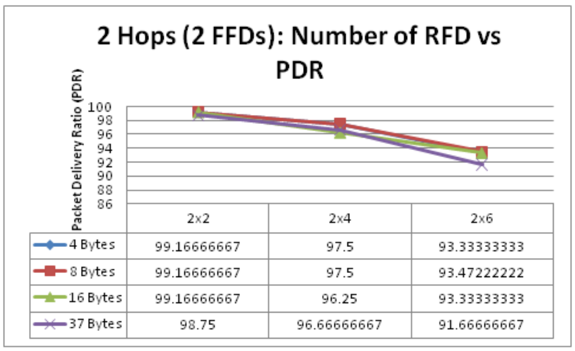

Figure 10. Performance of packet delivery rate against number of nodes for 2 hops

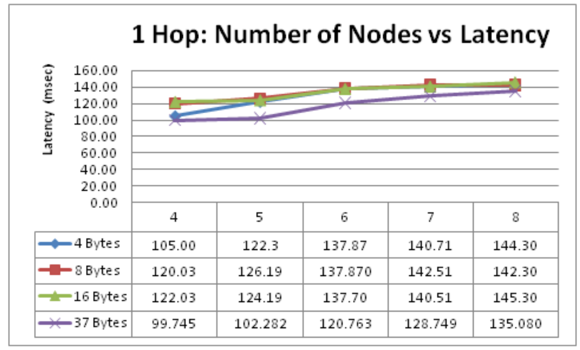

Figure 11. Average latency for 1 hop with various numbers of nodes with different packet sizes

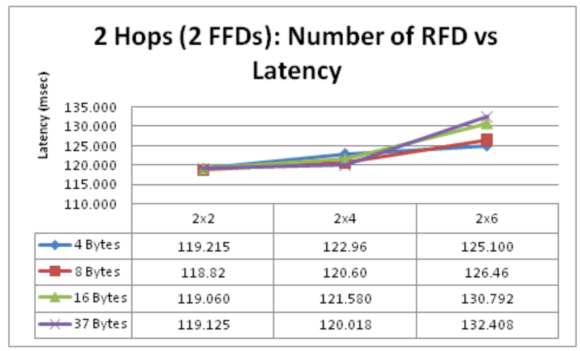

Figure 12. Average latency for 2 hops with various numbers of nodes with different packet sizes

Figure 9 shows the results of packet delivery rate against the number of nodes and data sizes. There are no changes with the increase of nodes for various packet sizes in 1 hop scenario. All the data transmitted was successfully received. However, with the increase in the number of hops, the packet delivery rate dropped as shown in Figure 10. It could be observed that the efficiency also dropped with the increase of packet size. This could be due to the relay node (Full Function Device(FFD))that was used could not handle the packets properly. It could also be because of the high number of operational access points which shares the same frequency as IEEE802.15.4 in the lab. To verify this, experiments were repeated outside the building, in an open space that does not have any access points coverage. The results obtained in outdoor environment were similar to the indoor environment. This proofs that the cause of packet drop is due to the relay node.

In order to calculate the latency, the client sends a ping message to a specific sensor node and round trip time (RTT) retrieved. From the RTT value, average per-hop latency is calculated. The average latency for 1 hop and 2 hops are given in Figure 11 and 12. The base latency which is latency measured with only one node active is about 65 milliseconds for 1 hop and 90 milliseconds for 2 hops. It can be observed that with the increase in the number of active nodes, the latency increased and ranges from 100 to 135 milliseconds. This average latency is comparable with average latency claimed in the white paper by IPSO-Alliance [10] which is about 125 milliseconds. Total latency is calculated based on the processing latency of packet at the node, processing latency at the network gateway or router and latency due to network condition. Major contribution of the latency is the wireless network condition such obstacles, interference of signal from other devices and others. We noticed that there is a slight decrease in latency when the experiments conducted in an open space without any other interference. Besides the two measurements, the average power consumption to transmit data every 20 seconds interval is about 0.015 volts. This was obtained by sending the available power in the battery as data. The difference between of power reading between 2 intervals were calculated and averaged. Based on the tests conducted, the average total consumption of battery before it malfunction is about 1.5 volts. This can be used to calculate the battery life of a sensor node in various types of implementations.

VI. Conclusion and Future Work

This paper proposed a gateway architecture to interconnect wireless sensor network with external network using

ISBN: 978-1-61208-207-3

6LoWPAN protocol. The gateway provides the mechanism for the end clients to directly communicate with the sensor node which was assigned with IPv6 address. Besides that the gateway forwards the periodical data to a web server.

The architecture is validated with the successful transmission of sensor data which was displayed using a client and web server. Further tests were conducted to validate the latency and the transmission success rate. The latency for 1 hop with various number of nodes ranges between 100 to 135 milliseconds while the transmission success rate is 100 % for 1 hop. The success rate drop with the increase of number of hop which could be because of the relay node (FFD) not forwarding the packets appropriately. Nevertheless the results are in accordance with the other prior art.

As future work, the proposed solution can be further tested in other environments by setting different transmission intervals, less interferences, etc. The performance can also be evaluated with the implementation of other components such as security, routing and mobility with multi-hop scenarios.

References

[1] IPv6 over Low Power Personal Area Network (6LoWPAN) IETF Working Group. Retrieved: July, 2012. http://datatracker.ietf.org/wg/6lowpan/

[2] G. R. Sinniah, Z. Suryady, U. Sarwar, M. Abbas, “A Gateway Solution for IPv6 Wireless Sensor Network”, Ultra Modern Telecommunication & Workshops, St. Petersburg, Russia. pp 1-6, October 2009.

[3] Y. Dun-Fan, M. Liang-liang, and W. Wei, “Design and Implementation of Wireless Sensor Network Gateway Based on Environmental Monitoring”, 2009 International Conference on Environmental Science and Information Application Technology, pp. 289-292, Jul. 2009.

[4] X. Jiang, S. Dawson-haggerty, P. Dutta, and D. Culler, “Design and Implementation of a High-Fidelity AC Metering Network”, IPSN ’09: Proceedings of the 2009 International Conference on Information Processing in Sensor Networks. Washington, DC, USA: IEEE Computer Society pp. 253-264, 2009.

[5] L. Wenbin, C. Dongxu, and Z. Junguo, “Design and Implementation of Wireless Sensor Network Gateway Faced to Forest Information Monitor”, 2010 International Conference on Intelligent System Design and Engineering Application, pp. 524-526, Oct. 2010.

[6] A. J. Jara, M. a. Zamora, and A. F. G. Skarmeta, “HWSN6: Hospital Wireless Sensor Networks Based on 6LoWPAN Technology: Mobility and Fault Tolerance Management”, 2009 International Conference on Computational Science and Engineering, pp. 879-884, 2009.

[7] D. Choi, J.-tak Park, and S.-yoen Kim, “IPv6 global connectivity for 6LoWPAN using short ID”, 2011 International Conference on Information Networking (ICOIN), 2011, pp. 384-387, 2011.

[8] A. Zimmermann and J. S. Silva, “6glad: IPv6 global to link-layer address translation for 6lowpan overhead reducing”, Next Generation Internet Networks, NGI 2008, pp. 209-214, 2008.

[9] “Sensinode hardware and NanoStack Operating System”, 2008. Retrieved: July, 2012. Available: http://www.sensinode.com/

[10] J. Abeill, M. Durvy, J. Hui, S. Dawson-Haggerty. “Lightweight IPv6 Stacks for Smart Objects: the Experience of Three Independent and Interoperable Implementations”, November 2008. Available at: http://www.ipso-alliance.org/white-papers

[11] “Network Stumbler”. Retrieved: July, 2012. Available: http://www.netstumbler.com/

[12] H. K. Jin, S. H. Choong and O. Koji, “A Routing Scheme for Supporting Network Mobility of Sensor Network Based on 6LoWPAN”, APNOMS ’07, LNCS 4773, pp. 155-164, 2007.

[13] ZigBee Alliance. “ZigBee Specifications”. Retrieved: July, 2012. http://www.zigbee.org/Specifications.aspx

[14] Hart Communication Foundation. “WirelessHART Overview”. Retrieved: July, 2012. http://www.hartcomm.org/protocol/about/aboutprotocol_specs.html

[15] The International Society for Automation. “ISA100, Wireless Systems for Automation”. Retrieved: July, 2012. http://www.isa.org/isa100

[16] Microchip Technology Inc. “MiWi Development Environment”. Retrieved: July, 2012. http://www.microchip.com/miwi

 ISBN: 978-1-61208-207-3