

# Performance Analysis of IP over IEEE 802.15.4 Radio using 6LoWPAN

Weijun Guo, [guow@cse.wustl.edu](mailto:guow@cse.wustl.edu) (A project report written under the guidance of [Prof. Raj Jain](#))



[Download](#)

---

## Abstract

IEEE 802.15.4 is the standard wireless link technology for low power, low cost, low data-rate, but long lifetime applications including wireless sensor network (WSN). Currently, whether or not IP architecture should be used in WSN over its 802.15.4 link is under dispute. One view is that IP architecture is not suitable for WSN; instead ad hoc network architecture is more preferred to be designed specifically for it. In contrast, IP architecture is seen more as an opportunity for WSNs, even some extra overhead is tolerable taking into consideration the benefits it will bring; and it is also argued that such overheads may actually be minimized through optimization. Such kind of arguments may last for a long time without some convincing experimental measurements of real case performances of IP over 802.15.4 implementation, and thus prevent the advance of related research. In RFC4944, IETF proposed the 6LoWPAN specification to enable IPv6 communication over low power, wireless personal area networks. Since then, we see a great demand and value of making an evaluation of this specification. In this paper, we carry out a performance evaluation of 6LoWPAN on the memory footprint, network capability, transmission delay, and routing cost using a Berkeley 6LoWPAN implementation, b6LoWPAN, on Tmote Sky running TinyOS 2.1. We try to alleviate the dispute and make the direction clearer to the community so as to promote further valuable research in this topic.

---

**Keywords:**6LoWPAN, Performance Analysis, IP over 802.15.4, Wireless Sensor Network

---

## Table of Contents

- [1.0 Introduction](#)
- [2.0 Experimental Design](#)
  - [2.1 Software](#)
  - [2.2 Hardware](#)
- [3.0 Performance Analysis](#)
- [4.0 Conclusion](#)
- [5.0 Future Work](#)
- [References](#)
- [List of Acronyms](#)

---

## 1.0 Introduction

Wireless sensor network (WSN) has been studied in both academia and industry for a score of years. Previous research topics focus on how to handle the technical difficulties in building and deploying such networks. MAC protocol, scheduling algorithm, radio model, interference, topology concern, power management, and etc. are the most popular topics in the community. With the maturity of sensor manufacturing techniques, more and more powerful sensors with more complex sensing capabilities have been made available. At the same time, more and more successful real environment WSN deployments have been carried out, for example

the Great Duck Island [Mainwaring02] for habitat monitoring, the Redwood tree monitoring [Selavo07] for environment monitoring. These deployments have accumulated good experience for sensor network construction.

Then the community starts to think of more complex scenarios of using WSNs, which include interoperability between different WSNs (important for cargo tracking and monitoring) and communication with other networks to extend the scope of physical information on the sensor nodes. Witnessing the success and ubiquitous of IP network, it is natural to try to figure out some approach to connect WSN to the existing IP network infrastructure, so that sensor networks deployed can be integrated into the IP family and take advantage of the existing network facilities, which have already matured and are available to public.

There are different ways to achieve this goal. One method is to design some middleware system that works as a bridge between different WSN gateways and also between outside IP networks. This approach limits interoperability only to gateways of the sensor network, which take all the responsibility of communication, and do not extend IP communication to the sensor nodes in the sensor network. A more interesting approach is to try to provide the ability of IP communication to all the nodes within the sensor network. By doing so, sensor nodes are more accessible to other network and can be fully used in different applications. The 6LoWPAN [Kushalnagar05, 6LoWPAN\_wiki] specification is one of this kind. It puts an adaption layer above the current 802.15.4 link layer and provides the ability of TCP/IP above this adaption layer as shown in Figure 1.

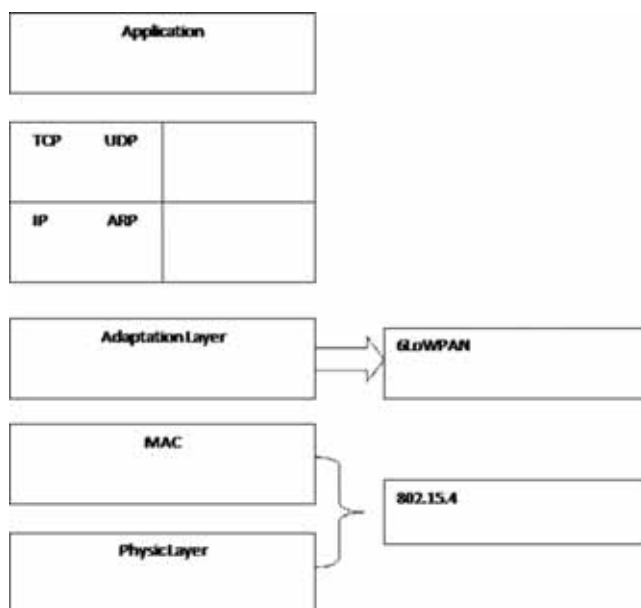


Figure 1: The position of 6LoWPAN in the network architecture

In this specification, every node in the wireless sensor network can implement this adaption layer and then be able to communicate with other IP end nodes. The adaptation layer is the main component of 6LoWPAN. It handles several things that such a connection of 802.15.4 to IP network needed. The first major function of this layer is the TCP/IP header compression. TCP/IP headers are too large for 802.15.4, which has a maximum packet size of 128 bytes; instead IPv6 header size is 40 bytes, UDP and ICMP header sizes are both 4 bytes, TCP header size is 20. Without compression, 802.15.4 is not possible to transmit any payload effectively. A second major function of the adaptation layer is to handle packet fragmentation and reassembling. 802.15.4 has a maximum frame size of 128 bytes, while IPv6 require for a minimum transmission unit (MTU) of 1280 bytes. This mismatch has to be handled in the adaptation layer. The third major function of the adaptation layer is routing. The border nodes of the WSN should be able to route IPv6 packets into the WSN nodes from outside and route inside packets to outside IP network. There are other functions of the adaptation layer on networking related things like neighbor discovery and multicast support.

While the principle of 6LoWPAN may be easy to understand, the implementation is not, considering the stringent constraints in processor speed, memory size and power on a sensor node. Wireless sensor nodes are of low power, low memory system but required for a long application lifetime (usually multi-years) with an 8 or 16-bit CPU of 8MHz, 48KB of ROM, 10KB of RAM, a maximum radio transmission speed of 250kbps, and powered by two AA batteries[Polastre05]. TCP/IP is originally designed for wired computers without such constraints; therefore, it does not take similar constraints into consideration in the design. This brings a lot of doubt on the notion of bringing TCP/IP to the wireless sensor network area instead of design ad hoc network architecture for it. Such concerns can be summarized as [Hui08]:

- Whether it is necessary to enable communication between sensor nodes to outside nodes?
- Is it practical to implement the TCP/IP architecture on resource constrained sensor motes?
- Would it be secure to do so?
- Does sensor node need identification like IP address?
- How much this would affect the performance of the sensing applications?

While these concerns are reasonable, they may underestimate the benefits of connecting a wireless sensor network to the IP network, which can be abbreviated as:

- It provides a capability to communicate with the mass majority of devices connected with IP. Today, essentially all computing devices use IP network stacks to communicate with other devices.
- The software, the tools, and the standards utilize IP and the layers above it, not the particular physical link underneath.
- The value of interoperability far outweighs the moderate overheads.

Furthermore, researchers have developed numerous mesh network layers over 802.15.4, as open source projects (such as TinyOS and micro-IP (uIP)), industrial forums (ZigBee and WirelessHART), or proprietary offerings (Dust Networks, Sencicast, and Millennial Net). Each has defined its own set of incompatible packet formats tied to particular MAC features, routing algorithms, and addressing. Many address only the individual 802.15.4 subnet, leaving all further communication protocols to be defined via ad hoc gateways. 6LoWPAN potentially lets us unify this disparate activity and enable embedded 802.15.4 devices to be incorporated into Ethernet, Wi-Fi, General Packet Radio Service, and other environments within a uniform IP framework.

Despite this, arguments may continue for a long time without enough performance analysis on the real case implementation was made. In this paper, we try to evaluate the performance of IP over 802.15.4 radio using the 6LoWPAN specification. We test the UDP, ICMPv6 protocols, and measure their throughput, delay and transmission error rate, and give the analysis of its performance the measured data may incline. We hope to give some useful clue in answering the doubts on how practical such an idea is to enable an IP capability for WSNs.

## 2.0 Experimental Design

### 2.1 Software

After the 6LoWPAN specification is made to public, Berkeley has implemented one open source version of this specification with most of its functions on TinyOS 2.1 [TOS\_Community]. Figure 2 shows the software architecture of their implementation. Similar to the standard TCP/IP architecture, their concerns focus on the layer 2 to layer 3 designs, which are the link layer and the network layer. They make full use of the current 802.15.4 capability and minimize the stack overhead in their implementation.

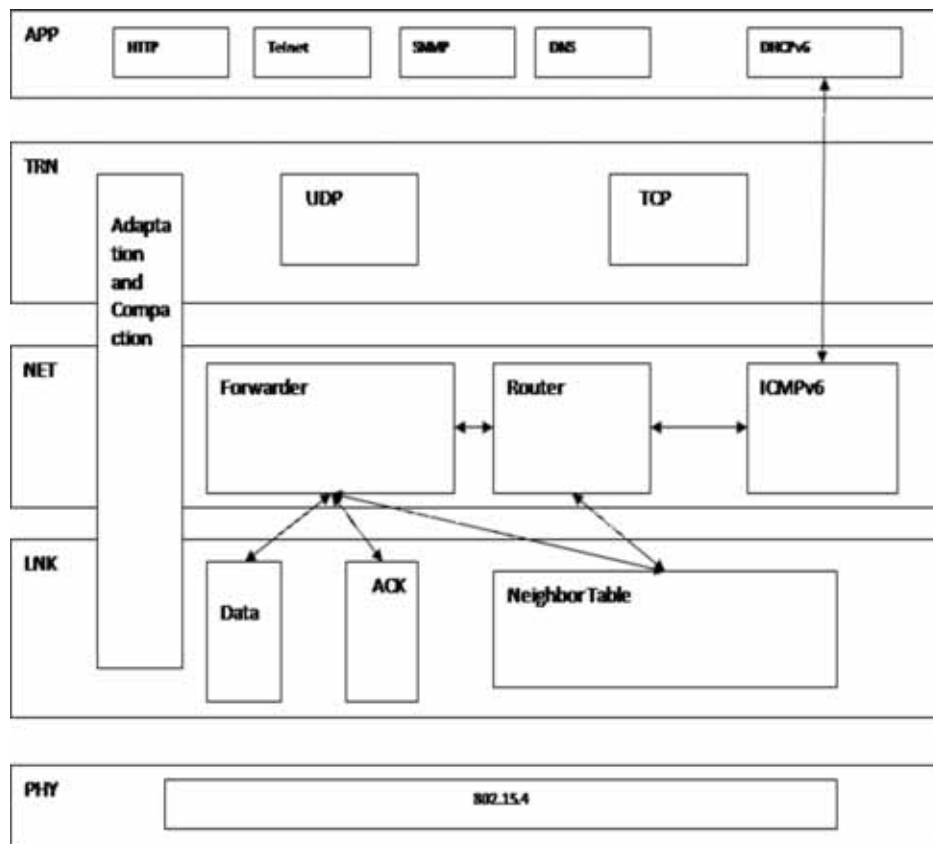


Figure 2: Software Architecture of 6LoWPAN

We base our experiments on this open source 6LoWPAN implementation, usually called b6LoWPAN. The experiments are done on TinyOS 2.1. There are three kinds of nodes in our experiments:

### 1. IPBaseStation

It is the border router of the whole sensor network and will collect all the performance data from the nodes inside the WSN. It also works as the bridge between the WSN and outside IP network, since they have different physical link layer and so have to do some modulation and demodulation in between. Usually, there can be one or more IPBaseStation for a wireless sensor network, so that they can distribute their responsibilities and provide fault tolerance;

### 2. IPMote

It runs the experimental application and transmits UDP, ICMP packets to other IPMotes or outside IP peers, and reports their performance data to the IPBaseStation for statistics analysis;

### 3. Relay node

It acts as relay hop within the WSN. Since it is typical to have multi-hop in real wireless sensor network deployment, we choose several representative multi-hop levels in our experiments respectively;

### 4. UDPEcho node

It provides only the UDP echo and ICMP echo function, to test the basic memory needed to support a workable 6LoWPAN enabled nodes.

## 2.2 Hardware

The hardware we use for our sensor motes is Tmote Sky using the CC2420 radio chip, which is 802.15.4 compliant. It has a radio transmission speed of 250 kbps and works on 2.4 GHz radio frequency. Figure 3 is an image of Tmote Sky we use in our experiments. It has temperature and humidity sensor, light sensor, solar sensor and extendable pins to install other sensor boards when needed.

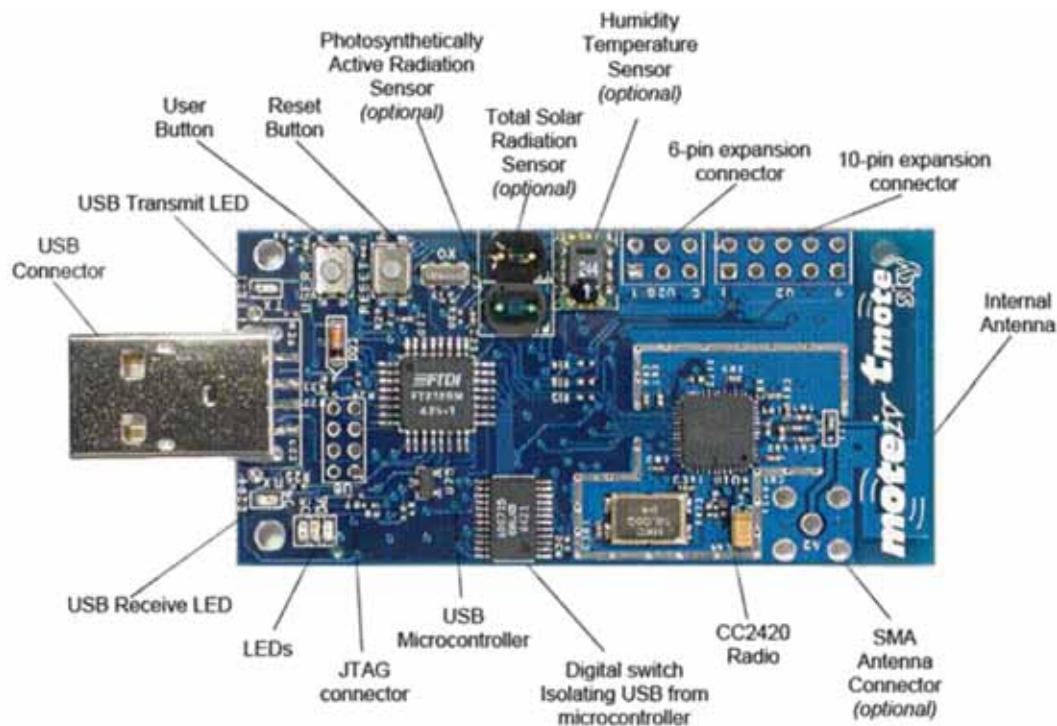


Figure 3: Tmote Sky

### 3.0 Performance Analysis

Our performance analysis focuses on several things.

1. Basic memory footprint of 6LoWPAN on sensor motes.

Here the metrics are the ROM and RAM footprint for having the basic 6LowPAN functions installed. This footprint is measured on the UDPEcho nodes we mentioned above. Except for this measurement, we also do a statistic of previous wireless sensor applications to evaluate whether the overhead of the memory footprint will interfere the installing of applications too much.

2. Average extra power consumption of 6LoWPAN on sensor motes.

We run our sensor network for a week and measure the power consumption on all the nodes in the sensor network. We use several representative duty-cycle workloads for such tests which are 0.1%, 0.5%, 1%, 5% respectively.

3. Average throughput of packet transmission within the sensor network using UDP.

Similar to the experiment 2, at this time we add UDP packet transmission between the sensor nodes. We start transmit UDP packets as soon as there is no application radio requirement to test the maximum and average throughput it can achieve without influence the existing application installed on the motes.

4. Average throughput of packet transmission between the sensor network and an outside IP nodes using UDP.

Similar to experiment 3, this time we do transmission between a randomly selected sensor node and also randomly selected IP node which is in the same physical link layer with the laptop the IPBaseStation is plugged in. We do not test any IP address far away, because it depends on too much other things not related with the performance of the 6LoWPAN we tested.

5. Average delay of packet transmission within the sensor network using UDP.

Similar to experiment 3, we measure the delay as another metric.

6. Average delay of packet transmission between the sensor network and an outside IP nodes using UDP.

Similar to experiment 4, here we use delay as the metric for evaluation.

7. Average retransmission needed for a transmission within the sensor network using UDP.

Wireless radio is not stable; sometimes it fails due to interference or low power level. We record the retransmission time use in the previous experiments and measure the average retransmission needed for a transmission.

8. Average retransmission needed for a transmission between the sensor network and an outside IP nodes using UDP.

Similar to experiment 7, this time the measurement is for transmission between a randomly selected sensor nodes and a IP node with the same physical link of the laptop the IPBaseStation is plugged in.

9. Average network transmission for routing establishment and maintenance within the sensor network.

Routing is a big part of the overhead of enabling an IP network layer in wireless sensor network. This experiment is designed to periodically send a signal to the IPBaseStation to get the routing information of the whole sensor network and record the packets and bits send in this process.

Due to the workload of constructing and debugging such a 6LoWPAN enabled sensor network, and carrying out the experiments, my work is not quite finished. To give some clue of what it will look like, I would recommend readers to read several papers [[Sikora05](#), [Mayer06](#), [Tolle07](#), [Christian05](#), [Dunkels07](#)], which provide some primary performance evaluation of 6LoWPAN.

In summary, a production-quality implementation that provides all of the functionality and support for one UDP socket and one TCP connection a) consumes 24,038 bytes of ROM and 3,598 bytes of RAM. These numbers include the entire runtime required to support the IPv6 network stack (e.g., OS-level services). b) The adaptation layer requires 6-11 bytes for a typical UDP/IPv6 header. c) The throughput for UDP is 9 kbytes/s for link-local and 1.7 kbytes/s over three hops. Throughput for TCP is 1.9 kbytes/s for link-local and 1.2 kbytes/s over three hops, which is significantly lower than UDP because TCP requires communication in both directions. d) The latency is measured by round-trip time of ICMP ping echo; and it is dependent on the MAC protocol and when the next channel sample occurs before the MAC can detect the ping packets.

As we may notice, the results shown in these papers are typical performance evaluations at the end of a conference paper in the research community. However, for a performance analysis paper, we need to do more complicate experimental designs and performance analysis. Such techniques will include those we learned from the course, like service definition, workload selection, metrics selection, fractional factorial design, confidence interval calculation, even regression model and other models applicable. All these



techniques will help us provide a more comprehensive and more convincing performance evaluation of IP over 802.15.4 radio wireless sensor network.

---

## 4.0 Conclusion

Our conclusions will be based on the previous experiments and performance analysis results to decide several crucial things for enabling IPv6 communication on 802.15.4 networks:

1. Whether the memory footprint is acceptable, which means it should not cause much limitation to the application of the sensor nodes to carry out its responsibilities in sensing.
2. Whether multi-hop communication within the sensor network works well and the maximum number of hops.
3. Whether network throughput both between the sensor nodes and between a sensor node and an outside IP peer node is satisfying in providing functionality, like connection detection, data collection, and other monitor and management control.
4. Whether network delay both between the sensor nodes and between a sensor node and an outside IP peer node will disturb normal human interaction.
5. Whether the extra power consumption is big enough to influence the lifetime of the sensor application too much.
6. Whether the cost of packet transmission for establishing and maintaining the routing for the whole sensor network exceeds some limit.

Overall, the experiments and performance analysis hope to answer the existing doubts and also give some clue of how much such a 6LoWPAN solution will fit for wireless sensor networks, and which areas should have more further research to give better performances.

---

## 5.0 Future work

As previously pointed out, b6LoWPAN is just one implementation of 6LoWPAN specification that is available to us at this time. In this implementation, there are some features that are not yet implemented in the package, like TCP transportation layer protocol, DHCPv6 for network auto-configuration, multicast support, and neighbor discovery. All these features can be further studied in the future when they become available.

Furthermore, there are also some other implementations of 6LoWPAN, including Nanostack [[NanoStack](#)], Arch Rock [[ArchRock](#)] 6LoWPAN implementation, and Sensinode [[Sensinode](#)] implementation. Though the last two implementations are both commercial ones, it would be interesting to do some performance comparison among these different implementations. For the performance data in our experiments, the reader should keep in mind that it is not the best (instead just the most initiative and open source available ones). So when making technique decision, you should know that the actual performance would be better than the data presented. This is also why a further comprehensive comparison of different implementations should be evaluated.

---

## References

1. [Mainwaring02] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk and J. Anderson, "Wireless sensor networks for habitat monitoring", In Proceedings of the ACM International Workshop on Wireless Sensor Networks and Applications, Sept. 2002.

2. [Selavo07] L. Selavo, A. Wood, Q. Cao, T. Sookoor, H. Liu, A. Srinivasan, Y. Wu, W. Kang, J. Stankovic, D. Young, and J. Porter, "LUSTER: Wireless sensor network for environmental research", In Proceedings of the ACM International Conference on Embedded Networked Sensor Systems (SenSys), Nov. 2007.
3. [Kushalnagar05] N Kushalnagar, G Montenegro, C Schumacher, "6LoWPAN: Overview, Assumptions, Problem Statement and Goals", IETF Internet Draft, 2005
4. [6LoWPAN\_wiki] Wiki page of 6LoWPAN, <http://smote.cs.berkeley.edu:8000/tracenv/wiki/b6loWPAN>, the wiki-site of 6LoWPAN
5. [Polastre05] J. Polastre, R. Szewczyk, and D. Culler, "Telos: enabling ultra-low power wireless research", In IPSN '05: Proceedings of the 4th international symposium on Information processing in sensor networks, page 48, Piscataway, NJ, USA, 2005. IEEE Press.
6. [Hui08] J. W. Hui and D. E. Culler, "Extending ip to low-power, wireless personal area networks", Internet Computing, IEEE, 12(4):37–45, July-Aug. 2008.
7. [TOS\_Community] tinoyos community, <http://www.tinyos.net>, the tinyos community and official site
8. [Sikora05] Sikora, A., "Implementing IP over Short Range Wireless Networks", IDAACS 2005. IEEE
9. [Mayer06] Karl Mayer and Wolfgang Fritsche, "IP-enabled wireless sensor networks and their integration into the internet", InterSense '06: Proceedings of the first international conference on Integrated internet ad hoc and sensor networks , Nice, France, 2006
10. [Tolle07] Gilman Tolle, "A 6LoWPAN application environment", SenSys '07: Proceedings of the 5th international conference on Embedded networked sensor systems, Sydney, Australia 2007
11. [Christian05] Andrew Christian, Jennifer Healey, "Gathering Motion Data Using Featherweight Sensors and TCP/IP over 802.15.4", IEEE International Symposium on Wearable Computing, Workshop on On-Body Sensing, 18-21 October 2005, Osaka, Japan
12. [Dunkels07] A. Dunkels, F. Österlind, and Z. He, "An adaptive communication architecture for wireless sensor networks", In SenSys '07: Proceedings of the 5th international conference on Embedded networked sensor systems, pages 335–349, New York, NY, USA, 2007. ACM.
13. [NanoStack] NanoStack. <http://sourceforge.net/projects/nanostack/>
14. [ArchRock] Arch Rock. <http://www.archrock.com/>
15. [Sensinode] Sensinode. <http://www.sensinode.com>




## Acronyms

WSN	Wireless Sensor Network
6LoWPAN	IPv6 Low-power Wireless Personal Area Network
RFC	Request For Comments
IETF	Internet Engineering Task Force
UDP	User Datagram Protocol
MTU	Minimum Transmission Unit

Last modified on November 24, 2008

This and other papers on latest advances in performance analysis are available on line at

<http://www.cse.wustl.edu/~jain/cse567-08/index.html>

    [Back to Raj Jain's Home Page](#)