

# A translation method between 802.15.4 nodes and IPv6 nodes

Shoichi Sakane    Yosuke Ishii    Katsuhiko Toba    Ken'ichi Kamada    Nobuo Okabe  
Yokogawa Electric Corporation  
{Shouichi.Sakane,Yosuke.Ishii,Katsuhiko.Toba,Ken-ichi.Kamada,Nobuo.Okabe}@jp.yokogawa.com

## Abstract

*There are many kinds of control networks based on non-IP, such as BA (Building Automation), FA (Factory Automation) and PA (Process Automation). The IPv6 and wireless technologies are expected to improve those networks. The IEEE 802.15.4 is a candidate for wireless technology in control networks because of its feature, e.g. low power consumption and small implementation. Seamless communication between both technologies is important, though it is not easy for IEEE 802.15.4 packet to carry IPv6 packet due to its limited features. This paper shows a translation method between IPv6 nodes and IEEE 802.15.4 nodes.*

## 1. Introduction

Control networks use various technically-different communication protocols like FOUNDATION Fieldbus [1] or BACnet [2]. Multiple protocols coexist within those networks. They are not based on the IP, but are going to make use of IP recently, FOUNDATION Fieldbus HSE, and BACnet/IP for example. IP-based control networks are deploying.

At the same time, wireless technology is expected to be useful in control networks. Because it can reduce their installation cost and increase their operational flexibility. A technology is required which can connect wireless networks with existing control networks seamlessly.

In this paper, there are two reasons why we propose a translation method between IEEE 802.15.4 nodes and IPv6 nodes. Firstly, the 802.15.4 meets the requirements of control networks, i.e. small size, low power consumption, and low cost. Secondly, the IPv6 will be the major network technology in control networks because it is cost effective and advancing rapidly.

We introduce characteristics of the 802.15.4 and the IPv6. Then we discuss mechanisms of the protocol translation to connect 802.15.4 nodes and IPv6 nodes with each, and propose a method of that.

## 2. Features of 802.15.4 and IPv6

This section describes the 802.15.4 and the IPv6 features related to the protocol translation.

### 2.1. Address

#### 802.15.4

In the 802.15.4, a communication domain is called PAN (Personal Area Network). A PAN has a two-octet identifier called PAN Identifier. A special 802.15.4 node that is responsible to packet delivering and dynamic address assigning is called Coordinator. 802.15.4 has two address types; Extended Address which is global unique, and Short Address which is dynamically assigned by a Coordinator.

#### IPv6

The IPv6 has a 128-bit address space. Higher 64 bits identifies a network, and it is distributed from routers. Lower 64 bits is usually generated from a unique identifier of the network interface on the node.

## 2.2. Node Identifier

### 802.15.4

An logical identifier of an 802.15.4 node is out of scope in the 802.15.4. It should be defined in the upper layer because Short Address is dynamically assigned. Extended Address can be a node identifier.

### IPv6

As an IPv6 address is not persistent as described above, a logical identifier is necessary. An FQDN are usually used as a logical identifier.

## 2.3. Packet Format

### 802.15.4

The maximum size of an 802.15.4 MAC frame is 127 octets. A MAC frame has a fixed 5 octets field, and the size of the address field varies from 4 to 20 octets depending on the purpose. The 802.15.4 allows to use an encryption algorithm using symmetric keys. It will consume up to 21 octets in conveying an encryption information.

To summarize, an 802.15.4 node can transfer data as its payload whose size is from 81 to 102 octets.

### IPv6

IPv6 header is 40 octets long. The maximum size of a payload is 65535 octets.

## 2.4. Packet Forwarding Mechanism

### 802.15.4

Coordinators correspond to forward packets. However the 802.15.4 does not define a certain mechanism to forward packets. It is needed to define any mechanism by other specification. For example, ZigBee [3] defines path control mechanisms based on AODV [4].

### IPv6

In IPv6 network, intermediate node called router forwards packets in a hop-by-hop fashion. Routers negotiate information with each to forward packets. There are various mechanisms like OSPF [5].

## 3. Translation Issues

In this section, we discuss the translation issues between 802.15.4 nodes and IPv6 nodes.

### 3.1. Identifiers of Nodes

A common logical identifier is necessary because there is no assurance that both IPv6 address and 802.15.4 address are not persistent. With logical identifiers, a node can identify other nodes independently of the protocols. A mechanism is necessary to resolve this identifier into a network (802.15.4 or IP) address.

### 3.2. Translation Method of Packets

Similarly, there are two possibilities on how to deliver a packet from an 802.15.4 node to an IPv6 node.

- A) A virtual 802.15.4 address is assigned to a destination IPv6 node.
- B) An 802.15.4 node explicitly sends a packet to the translator, and the translator delivers it to the IPv6 node.

There are two possibilities on how to deliver a packet from an IPv6 node to an 802.15.4 node.

- C) A virtual IPv6 address is assigned to a destination 802.15.4 node.
- D) An IPv6 node explicitly sends a packet to the translator, and the translator deliver it to the 802.15.4 node.

With the A) or C) scheme, an originating node does not need to distinguish the peer's protocol. That is, an 802.15.4 node can communicate with IPv6 nodes transparently. In other word, an node can not know the peer's protocol. The feature of B) or D) is the opposite of the above.

### 3.3. Payload size

a gap of payload size, i.e. maximum 102 octets of an 802.15.4 packet and maximum 65535 octets of an IP packet, must be considered. One method is that the translator fragments and reassembles packets to fill in the gap. Another method is that nodes should be designed to prevent any of fragmentation or reassembling.

## 4. A Proposal of a Translation Method

In this section, we propose a translation method that enables the communication between 802.15.4 nodes and IPv6 nodes based on above discussion.

### 4.1. Identifiers of Nodes

We introduce the device identifier (*devid*) to identify an 802.15.4 node or an IPv6 node in a similar way. An application uses *devids* to identify the peer nodes transparently.

In IPv6 network, existing directory services (LDAP [6], etc) can be used for this name resolution.

In 802.15.4 network, we introduce the dynamic configuration aggregator (DCAGGR) because there is no existing infrastructure for name resolution. DCAGGR provides a service like dynamic DNS, which means that an 802.15.4 node registers its address and *devid* to DCAGGR, and other nodes can resolve the *devid* into the address.

As the *devid* is an effective means to abstract nodes, we use it in communication between native 802.15.4 nodes as well. For this reason, we introduce an application header, which specifies both source and destination *devids*, at the top of an application payload.

The *devid* resolution functions in the both networks should be logically separated because of the following reasons.

- To make the framework of *devid* and DCAGGR be functional even on standalone 802.15.4 networks.
- Not to interfere with evolution of both networks.
- To save 802.15.4 network resources by preventing IPv6 nodes from querying to DCAGGR.

### 4.2. Translation Method of Packets

We use the method A) in section 3.2 to translate packets from 802.15.4 nodes to IPv6 nodes. The reason is that switching the behavior corresponding to the type of the peer is not suitable for an 802.15.4 node because an 802.15.4 node has a limited resource, and the translation between 802.15.4 nodes and other non-IP nodes may be required because non-IP control networks can not be ignored due to long lifetime of control networks. Therefore, seamless communication mechanism is required for 802.15.4 nodes.

The proposed translation method assumes that there is a packet forwarding mechanism in 802.15.4 network, and the translator can resolve an 802.15.4 address assigned to the IPv6 node. So that a packet whose destination is an IPv6 node should be delivered to the translator. The translator resolves the 802.15.4 address into the corresponding IPv6 address. After that, the packet is delivered to the destination IPv6 node by the delivery mechanism of IPv6 network.

We use the method D) in section 3.2 to translate packets from IPv6 nodes to 802.15.4 nodes. The application in the IPv6 network should be designed to prevent that an 802.15.4 node reassembles packets. The application in the 802.15.4 network should be also designed to prevent the node from fragmenting packets. Because there is no benefit for applications in the 802.15.4 network to process large packets, and 802.15.4 nodes may not be able to process fragmented packets due to its limited capabilities. Therefore, network transparency has less meaning for IPv6 nodes. In other point of view, the translator should have a counter measures against large IPv6 packets to save resources of 802.15.4 nodes.

We assume that there is a kind of a directory service or a service discovery to obtain the translator's IPv6 address. An IPv6 node sends a packet to the translator explicitly and the packet is delivered to the translator by the delivery mechanism of IPv6 network. The translator receives a packet, resolves the destination *devid* into the 802.15.4 address, and sends it to the 802.15.4 address. And then, the packet is delivered to the destination 802.15.4 node by the delivery mechanism of 802.15.4 network.

### 4.3. Overview of the proposal

The overview is shown in Figure 1. An 802.15.4 node **A** has a *devid* **Da** and an 802.15.4 address **Wa**. An IPv6 node **B** has a *devid* **Db**, an IPv6 address **IPb**, and a virtual 802.15.4 address **Wb**. The translator has an IPv6 address **IPt**. DCAGGR is used to resolve *devids* and 802.15.4 addresses. Coordinator manages virtual addresses to be assigned to IPv6 nodes.

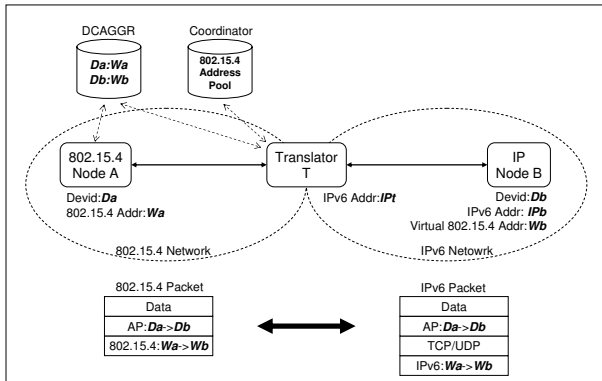


Figure 1. Overview of the Translator

Figure 2 shows the diagram of the translation from the 802.15.4 node **A** to the IPv6 node **B**.

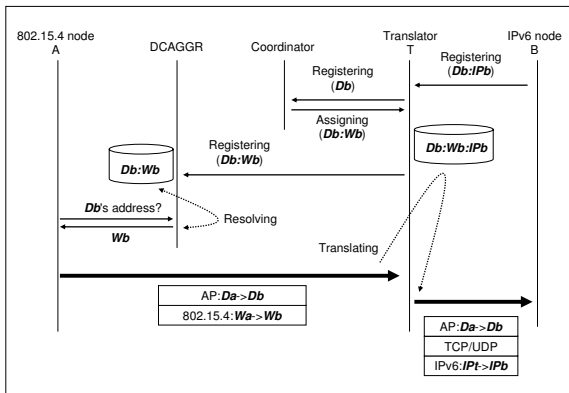
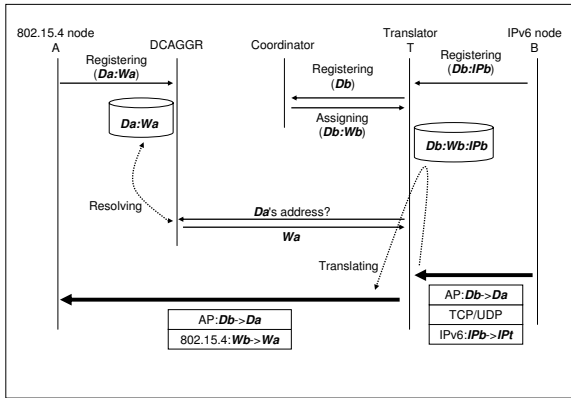


Figure 2. Translation from 802.15.4 to IP

1. **B** informs **T** of its *devid* **Db** and IPv6 address **IPb**.
2. **T** requests Coordinator with **Db** to get an address **Wb**.
3. Coordinator picks up an address **Wb** from the address pool, and sends it back to **T**.
4. **T** registers **B**'s *devid* **Db** and the virtual 802.15.4 address **Wb** to DCAGGR.
5. **A** asks the virtual 802.15.4 address **Wb** to DCAGGR.
6. **A** sends an 802.15.4 packet for **B** to **Wb**, and it arrives at **T**.
7. **T** knows that **B** is an IPv6 node and **B** has **IPb**, translates the 802.15.4 packet to an IPv6 packet, and sends the packet to **IPb**.



**Figure 3. Translation from IPv6 to 802.15.4**

Figure 3 shows the diagram of the translation from the IPv6 node **B** to the 802.15.4 node **A**. It assumes that the step of #1 to #4 of Figure 2 has been done.

1. **A** registers its *devId* **Da** and address **Wa** to DCAGGR.
2. **B** obtains the *devId* **Da** of the peer **A**. This information may be embedded in the configuration of the application retrieved from a directory service.
3. **B** sends a packet to **T** which will be forwarded to **A**. The packet contains *devId* **Da** of **A**.
4. **T** asks the *devId* **Da** to the DCAGGR, and obtain the destination 802.15.4 address **Wa**.
5. **T** resolves **Wb** from a registry.
6. **T** converts the IPv6 packet into an 802.15.4 packet and sends it to **Wa**.

## 5. Future Work

### Management

There is a requirement to figure out the state of a wireless network including devices correctly. In the IPv6 network, heart beats of ICMP messages are used to check reachability. But it is not suitable for wireless network because devices in wireless network usually work on batteries. The translator should manage such confirmation packets from an IPv6 network.

### Redundancy

DCAGGR and the translator is a single point of failure. We need to improve this. For example, making them multiplex.

### Communication between multiple 802.15.4 networks

In this paper, we assume that a single 802.15.4 network is connected with the IPv6 network through the translator. The IPv6 network can cover broader geographical area than 802.15.4 network. There may be a need for multiple 802.15.4 networks to be connected with the IPv6 network.

### Security

For the security of the IPv6 network side, an IPv6 packet can be forged, and a packet can be eavesdropped. Per-packet security is necessary to avoid this. In particular, a malicious IPv6 node can use a false *devId* in this proposed method. It can lead to attack such as exhausting the pool of reserved *devIds*, or spoofing *devIds* used by others. The translator must authenticate IPv6 nodes and verify whether the correct *devIds* are used. We are developing a security mechanism suited to IP-based control networks [7]. We are examining the solution of those security issues by applying the mechanism.

For the security of 802.15.4 network, there are various on-going researches on it. We need to examine them and research how to apply them to our proposed method.

## 6. Conclusions

In this paper, we described both characteristics of the IEEE 802.15.4 that would be suitable for control networks, and the IPv6 that would be one of major protocols in the Internet. We discussed the issues that would happen when the nodes communicated with each, and proposed a method to solve them.

## References

- [1] Fieldbus Foundation, “FF-581-1.3, FOUNDATION Specification: System Architecture”, 2003.
- [2] ASHRAE, “ANSI/ASHRAE Standard 135-1995, BACnet A Data Communication Protocol for Building Automation and Control Networks”, 1995.
- [3] Zigbee Alliance, <http://www.zigbee.org/>.
- [4] C. Perkins, E. Belding-Royer, and S. Das, “Ad hoc On-Demand Distance Vector (AODV) Routing”, RFC 3561, Jul 2003.
- [5] J. Moy, “OSPF Version 2”, RFC 2328, STD 54, Apr 1998.
- [6] M. Wahl, T. Howes, and S. Kille, “Lightweight Directory Access Protocol (v3)”, RFC 2251, Dec 1997.
- [7] N. Okabe, et al, “A Prototype of a Secure Autonomous Bootstrap Mechanism for Control Networks”, SAINT 2006, Jan 2006.