Linux-based 6LoWPAN border router

David Hauweele

University of Mons

7 August 2013

Table of Contents

1 Internet of Things

2 Problem and state of the art

3 Implementation

4 Validation

5 Conclusion

Internet of Things

Internet today

- Not only interconnected computers
- Mobile Internet (smartphones, tablets)
- Interconnected objects (sensors, automation, monitoring)

Internet of Things

Internet today

- Not only interconnected computers
- Mobile Internet (smartphones, tablets)
- Interconnected objects (sensors, automation, monitoring)

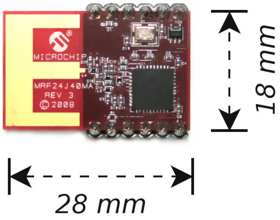
Definition

- The **Internet of Things** is a concept which seeks to integrate physical objects to Internet with an adressing system which can uniquely identify them.
- These objects are generally equipped with *sensors* and *actuators* in order to interact with their environment and they have *limited* processing capabilities $\rightarrow$ *smart objects*.

Wireless Sensor Network

Features

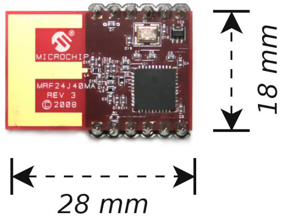
- Wireless nodes
- Tiny
- Cheap ($\approx$ 5 – 75 € $\rightarrow$ 650 – 10000 ¥)
- Low power ($\approx$ 20 – 25 mA)



Wireless Sensor Network

Features

- Wireless nodes
- Tiny
- Cheap ($\approx$ 5 – 75 € $\rightarrow$ 650 – 10000 ¥)
- Low power ($\approx$ 20 – 25 mA)



Problems

- Proprietary solutions
- A lot of different solutions
- Lack of interoperability

Need standard protocols

Standardization

How to connect these wireless nodes to Internet ?

- Physical and link layer $\rightarrow$ IEEE 802.15.4
- Network layer $\rightarrow$ ZigBee or network IP (IPv6)
- Upper layers $\rightarrow$ dedicated framework or TCP/UDP/ICMP

Source: http ://compixels.com

Standardization

How to connect these wireless nodes to Internet ?

- Physical and link layer → IEEE 802.15.4
- Network layer → ZigBee or network IP (IPv6)
- Upper layers → dedicated framework or TCP/UDP/ICMP

We use IPv6 on top of the IEEE 802.15.4 standard

Source: http ://compixels.com

IEEE 802.15.4

IEEE 802.15.4

The **IEEE 802.15.4**[1] standard defines the physical layer (PHY) and media access control (MAC) for low-rate wireless personal area networks. In particular, this standard is optimized to let smart objects communicate with each other.

- Low power $\rightarrow$ autonomy
- Data rate $\approx$ 20 – 250 kbits/s
- Small frames (127B)
- Optimized for wireless sensor networks
- Use the license-free band ISM (783 / 868 / 915 MHz – 2,4 GHz)

6LoWPAN

Network layer IPv6

- Very large number of adresses ($2^{128} \approx 3,4 \times 10^{38}$)
- Directly exposed to Internet

Problems

The IPv6 network is optimized for an use with efficient links with low loss-rate (Ethernet, Wi-Fi, ...). This poses harsh contraints on the nodes used within a wireless sensor network.

- Large addresses make large headers $\rightarrow$ IPv6 header is 40B
- MTU of 1280B $\rightarrow$ IEEE 802.15.4 frame is 127B

6LoWPAN

Solution → 6LoWPAN [8]

- Adaptation layer between IPv6 and IEEE 802.15.4
- Adjust the IPv6 network to the performances of smart objects

6LoWPAN

Solution → 6LoWPAN [8]

- Adaptation layer between IPv6 and IEEE 802.15.4
- Adjust the IPv6 network to the performances of smart objects

Provides

- Compression of the IPv6 and UDP headers
- Multiple compression schemes for IPv6 addresses
- Transparent fragmentation of 1280B packets to fit 127B frames
- Optimization of the IPv6 Neighbor Discovery Protocol

Example of node

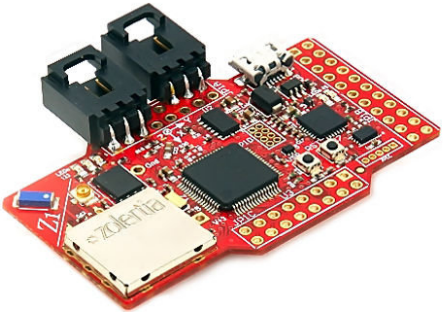


FIGURE : Zolertia Z1

Source: http ://www.zolertia.com

Example of node

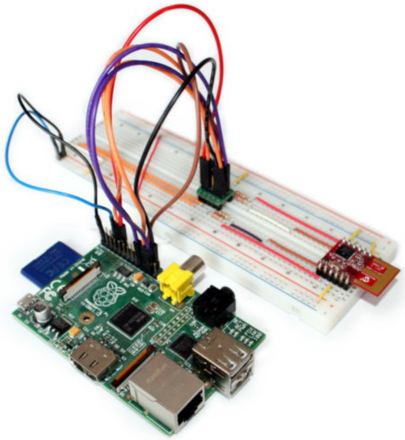


FIGURE : Raspberry Pi and MRF24J40

Table of Contents

1 Internet of Things

2 Problem and state of the art

3 Implementation

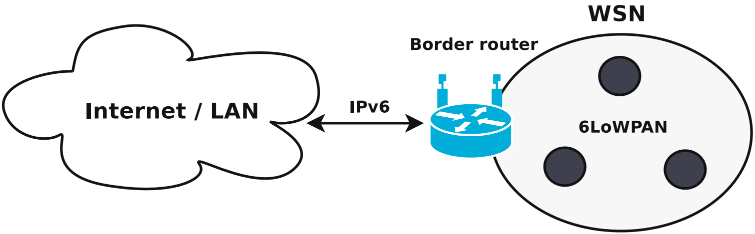
4 Validation

5 Conclusion

Border router

Border router

The **border router** is a special equipment of the wireless sensor network. It allows the coordination of the network and provides a gateway to the external world.



Border router

Roles of the border router

- Forwarding between IPv6 and 6LoWPAN
- IP configuration of the nodes (6LoWPAN-ND [9])
- Multihop routing (RPL [10])
- Context sharing

Existing solutions

Existing 6LoWPAN border routers

- Grinch [7]
- Arch Rock PhyNet Router [3]
- NanoRouter [4]
- JenNet-IP Border-Router [2]
- 6LBR by the CETIC [6]

Existing solutions

Existing 6LoWPAN border routers

- Grinch [7]
- Arch Rock PhyNet Router [3]
- NanoRouter [4]
- JenNet-IP Border-Router [2]
- 6LBR by the CETIC [6]

Linux based solutions

Some of these solutions are based on Linux. However they do not use a solution implemented directly into the Linux Kernel.

Existing solutions

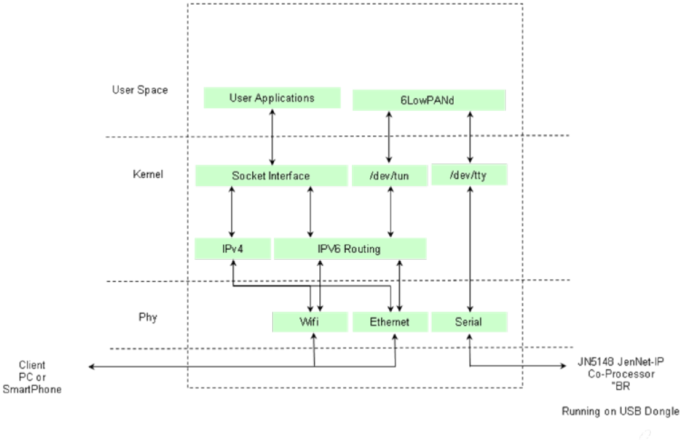


FIGURE : Architecture of the JenNet-IP Border-Router [2].

Linux

The Linux-ZigBee project

- Support IEEE 802.15.4 and 6LoWPAN
- Directly into the Linux Kernel
- Support some IEEE 802.15.4 transceivers
- Some features missing
- Some bug remains

Project goals

Goal

Realization of a **border router** on **Linux** based on the IEEE 802.15.4 and 6LoWPAN (**Linux-ZigBee**) subsystem on a cheap and low power embedded platform supporting Linux, the **Raspberry Pi**.

Project goals

Goal

Realization of a **border router** on **Linux** based on the IEEE 802.15.4 and 6LoWPAN (**Linux-ZigBee**) subsystem on a cheap and low power embedded platform supporting Linux, the **Raspberry Pi**.

Interests

- Development, testing and debugging of the Linux-ZigBee project
- Lessen the dependency on specific solutions
- Solution based mainly on the kernel
- Can be extended to numerous platforms

Table of Contents

1 Internet of Things

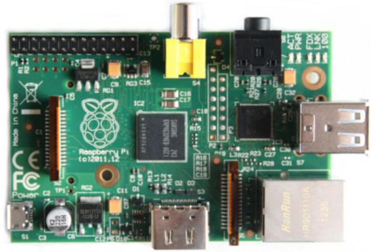
2 Problem and state of the art

3 Implementation

4 Validation

5 Conclusion

Raspberry Pi



Installation

- Cleaning of the Raspbian distribution
- Configuration and optimization of the tools
- Cross-compilation
- Creation of backup tools for the SD card containing the OS
  $\rightarrow$ reduce the time and size of backups with a factor $\approx$ 20 and 40

Backporting

Backporting

The Linux-ZigBee project was not available in the kernel version used on the Raspberry Pi at the time (3.2.27). Since it wasn't possible to directly use a more recent version, it was necessary to backport the changes from the development version.

Problems

- 3.2.27 is very old
- API changes
- Some subsystems didn't exist

Debugging tools

WSN-Tools

In order to ease debugging, we have developped new tools. The project **WSN-Tools** [5] contains tools for use in IEEE 802.15.4 Wireless Sensor Networks. These tools allow the manipulation of MAC 802.15.4 frames, setting up a sniffer, injecting and replaying traffic directly from the command line.

Interests

- Ease debugging of IEEE 802.15.4 and 6LoWPAN networks
- Record traces for later validation
- Live inspection of the 802.15.4 traffic
- Build customized frames

Bugs fixed in the Linux-ZigBee project

Transceiver's driver (MRF24J40)

- Missing interruption
- Reception during transmission
- Impact of the kernel architecture of the Raspberry Pi
- Duplicated packets

6LoWPAN layer

- IPHC link-local addresses compression/decompression
- 6LoWPAN fragmentation

Additional features

Additional features

- Audit driver → performances measurements and testing
- Turbo mode (625Kbps)
- Support for the 3.8 kernel

Table of Contents

1 Internet of Things

2 Problem and state of the art

3 Implementation

4 Validation

5 Conclusion

Validation

Tests

- Simulation of ping
- Ping (normal, flood and broadcast)
- UDP and TCP traffic
- Fragmentation (ICMP and UDP)
- Speed measurements
- Benchmarking the driver
- Static routing
- Address autoconfiguration with radvd
- Gateway IPv6/6LoWPAN
- Interaction with a node using the RTOS Contiki

Fragmentation

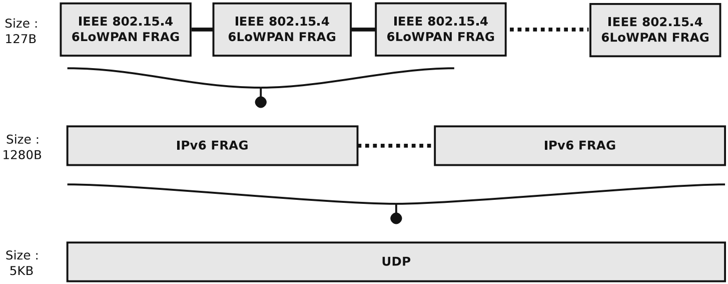


FIGURE : Fragmentation of an 5KB UDP datagram.

Fragmentation and speed

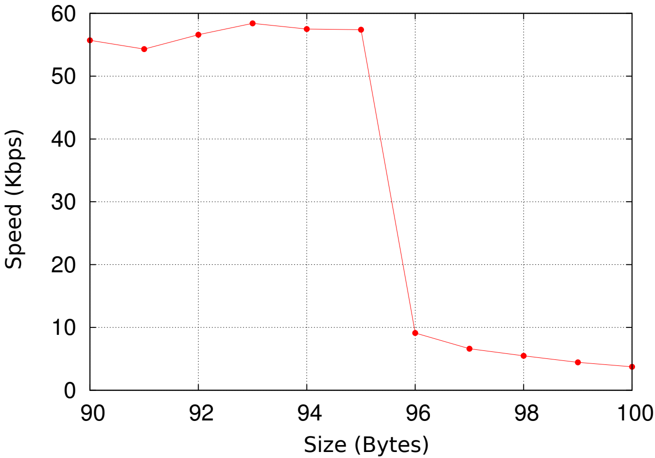


FIGURE : Speed according to the size of the packet.

Table of Contents

1 Internet of Things

2 Problem and state of the art

3 Implementation

4 Validation

5 Conclusion

Conclusion

- New approach for the development of a 6LoWPAN border router :
  1. Based entirely on Linux
  2. Use the new IEEE 802.15.4/6LoWPAN subsystem in the Linux Kernel
  3. Fonctionnality tests and performance measurements of this subsystem
  4. Correction of some bugs in this subsystem
  5. One of the first working solution of this type

Conclusion

- Debugging tools for Wireless Sensor Networks
- Familiarization with Linux kernel development
- Some patches are on their way to the Linux Kernel
- Offer a new solution for the development of a 6LoWPAN border router on Linux

Slides

http://www.hauweele.net/~gawen/files/6lowpan.pdf

References I

- IEEE Standard for Local and metropolitan area networks - Part 15.4 : Low-Rate Wireless Personal Area Networks (LR-WPANs). *IEEE Std. 802.15.4^{TM}- 2011*, 2011.
- JenNet-IP border-router. JN-AN-1110 (Application Note), 2011.
- Arch Rock. `http://www.archrock.com`, 2012.
- Texas Instrument CC-6LoWPAN. `http://processors.wiki.ti.com/index.php/CC-6LoWPAN`, 2012.

References II

WSN-Tools.
`http://www.hauweele.net/~gawen/wsn-tools.html`, 2013.

M. Denis.
Rapport de stage CETIC : Prototype de border router Redwire Econotag sous Contiki.
2012.

Maciek.
Grinch - simple 6lowpan RPL border router.
`http://sixpinetrees.blogspot.be/2011/06/grinch-simple-6lowpan-rpl-border-router.html`, June 2011.

References III

G. Montenegro, N. Kushalnagar, J. Hui, and D. Culler.
Transmission of IPv6 packets over IEEE 802.15.4 networks.
RFC 4944 (Draft Standard), 2007.

Z. Shelby, S. Chakrabarti, and E. Nordmark.
Neighbor discovery optimization for low power and lossy networks (6LoWPAN).
Internet-Draft ”draft-ietf-6lowpan-nd-21.txt”, Aug. 2012.
(Work in progress).

T. Winter, P. Thubert, A. Brandt, J. Hui, R. Kelsey, P. Levis, K. Pister, R. Struik, J. Vasseur, and R. Alexander.
RPL : IPv6 routing protocol for low power and lossy networks.
RFC 6550 (Draft Standard), Mar. 2012.