OpenWSN & OpenMote: Demo'ing A Complete Ecosystem for the Industrial Internet of Things

Tengfei Chang, Pere Tuset, Jonathan Munoz, Xavier Vilajosana, Thomas Watteyne

OpenWSN & OpenMote: Demo'ing A Complete Ecosystem for the Industrial Internet of Things

Tengfei Chang*, Pere Tuset-Peiro[†‡], Xavier Vilajosana[†‡], Thomas Watteyne*

*Inria-Paris, EVA Team, France

[†]OpenMote Technologies, Barcelona, Spain

[‡]Universitat Oberta de Catalunya, Barcelona, Spain

Abstract—The Industrial Internet of Things (IIoT) vision relies on reliable low-power wireless technologies and a complete standardized protocol stack that brings Internet connectivity to constrained end devices. Currently IEEE802.15.4-2015 TSCH provides reliability over low-power wireless technologies and the IETF has defined a protocol stack that is suitable to constrained devices. Open-source initiatives contributing to the development of network stacks for embedded devices, and the availability of open hardware platforms that facilitate prototyping IIoT applications contributes to the adoption of IIoT technologies.

To realize the IIoT vision, this demo presents the Open-WSN and OpenMote projects, which provide both an open-source software implementation of the IEEE low-power wireless technologies and the IETF protocol stack, and an open-source hardware platform that meets the requirements to prototype IIoT applications.

Keywords- OpenWSN, OpenMote, 6TiSCH, Industrial IoT.

I. Overview

The Industrial Internet of Things (IIoT) brings together information and operations technologies to enable novel monitoring and control applications in the industry. The IIoT vision relies on low-power wireless technologies to reduce network deployment time and cost, and on standard Internet protocols to enable end-to-end communication between the operation center and leaf devices. Until today, two main challenges have stopped the IIoT vision from becoming a reality. First, low-power wireless technologies did not meet the small energy consumption and high reliability requirements of industrial applications, which are taken for granted in legacy deployments using wired technologies. Second, the Internet protocol stack that allows end-to-end communications was not adapted to the requirements of constrained leaf devices that are typically found in these environments.

Over the last years, different efforts have allowed to overcome these challenges. Regarding reliability, the IEEE802.15.4-2015 standard for low-power wireless communications includes TSCH (Time Synchronized Channel Hopping), a MAC (Medium Access Control) technology that relies on time synchronization to achieve <50 uA current consumption and channel hopping to achieve over 99.999% end-to-end reliability. Regarding end-to-end connectivity, the IETF (Internet Engineering Task Force) has defined a protocol stack for constrained devices that includes 6LoWPAN (RFC4919 [1], RFC6282 [2]) as the IPv6 protocol adaptation layer to IEEE802.15.4 networks, RPL (RFC6550 [3]) as the
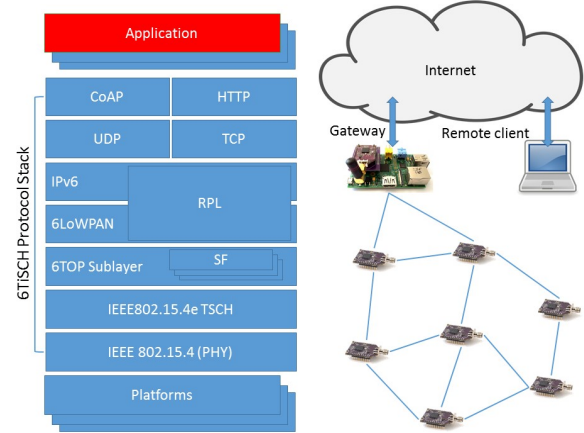

Fig. 1. OpenWSN network architecture.

routing protocol for LLN (Low-Power Lossy Networks), and CoAP (RFC7252 [4]) as a light-weight HTTP-like application protocol.

However, adoption of IIoT technologies and standards still has a long way to go mainly due to the lack of open-source implementations that foster their adoption and ensure interoperability. In that direction, the OpenWSN[1] project was founded in 2010 in the Berkeley Sensor & Actuator Center (BSAC) at UC Berkeley to provide an open source implementation of the IEEE/IETF protocol stack. Over the last years, the OpenWSN stack has been ported to various hardware architectures and platforms, including legacy boards such as the TelosB, and modern platforms such as the IoT-Lab boards.

Unfortunately, these boards do not meet the processing, memory and low-power requirements of modern industrial monitor and control applications, or are hard to obtain commercially. This represents another barrier to IIoT adoption, since prototyping of applications becomes difficult and expensive. For that reason, members of the OpenWSN team founded OpenMote Technologies[2] in 2014 with the aim to provide an open hardware ecosystem for the IIoT that is powerful, easy-to-use and affordable. The combination of both projects aims to provide the tools required to kick-start adoption and realize the IIoT vision.

---

[1] http://www.openwsn.org/

[2] https://www.openmote.com/

The remaining of this extended abstract presents the standard protocol stack for the IIoT, summarizes the most relevant aspects of the OpenWSN and OpenMote projects, and presents the IIoT demo that is being showcased.

II. Standardizing the Industrial IoT

The IEEE802.15.4-2015 [5] TSCH mode defines a TDMA (Time Division Multiple Access) MAC layer protocol which divides time into timeslots. Each packet transmission in a TSCH network happens in a timeslot, at a specified frequency. All devices in the network are synchronized by aligning their timeslot boundaries and sending packets on different channels (channel hopping). With the same sense of time, devices can efficiently optimize the usage of the radio, which is the major energy consuming part on board, to keep energy consumption low. With frequency diversity, devices can effectively reduce the influence of external interference and multi-path fading to improve the reliability of network [6].

Given the industrial-grade performance of TSCH, the IETF 6TiSCH Working Group [7] is now focusing on adapting the IPv6 protocol on top of it, to achieve a reliable and Internet-enabled IIoT protocol stack. The protocol stack is composed of 6LoWPAN as the IPv6 adaptation layer for IEEE802.15.4-based networks, RPL as the routing protocol for LLN, and CoAP as a light-weight HTTP-like application protocol. Additionally, the IETF 6TiSCH WG has created the 6top Protocol (6P) to glue TSCH and the IETF upper stack together. In particular, 6P is a pairwise slot allocation protocol used to dynamically build a network schedule depending on the communication requirements to each neighbour and to the root Different allocation policies, named Scheduling Functions, are open to developers who can build their network according to their specific demands, i.e., bandwidth or latency.

Overall, the 6TiSCH protocol stack enables industrial-grade communications to be fully Internet-enabled, and therefore facilitates the integration of information and operation technologies to build end-to-end sense and control applications.

III. OpenWSN: a standard-based open source implementation for IIoT

OpenWSN [8] is an open source project that has become the reference implementation of a standards-based protocol stack for the IIoT (Fig. 1), including support for IEEE802.15.4-2015 TSCH, 6LoWPAN, RPL and CoAP, as described earlier.

OpenWSN is organized into two sub-projects: firmware and software. The firmware sub-project is the code that runs in embedded devices. It is written in standard C99 and uses the GCC as the default compiler, which provides an open-source toolchain for the supported architectures. In particular, it has been ported to 3 different processor architectures (AVR, MSP430 and ARM Cortex) and 11 different hardware platforms[3], from the legacy TelosB board to the state-of-the-art OpenMote platform (described in detail in the next section). In turn, the software sub-project (named OpenVisualizer) is
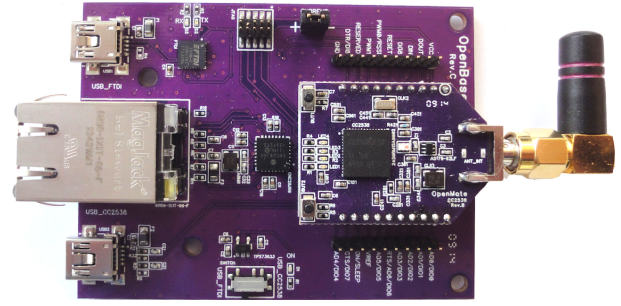

Fig. 2. An OpenMote-CC2538 and an OpenBase board.

the code that runs on a computer and serves as a gateway to the Internet. It is written entirely in Python and is responsible for 6LoWPAN/IPv6 compression and RPL downstream routes, as well as network topology and statistics, i.e., packet delivery metrics (RSSI, LQI, ETX) and routing topology. Finally, documentation for the firmware and the software sub-projects is compiled using Doxygen.

To manage the firmware, software and documentation build process OpenWSN utilizes SCons, a Python-based construction tool. Command-line options allow users to compile hundreds of different projects by combining boards, application, toolchains and kernel, with a single command line. In addition, CI (Continuous Integration) servers[4] allow to automatically compile and build all projects for all platforms, enabling non-regression testing and allowing end users to build their applications online, thus acting as an open IIoT service.

The OpenWSN project also provides a set of tools that focuses on easing development and prototyping of IIoT applications, allowing users to deploy an OpenWSN network within minutes. On the one hand, OpenPi is a Raspberry Pi image based on Raspbian that has all the software tools required to build and deploy an OpenWSN network. On the other hand, OpenVM is a virtual machine build using Vagrant that provides a complete development environment for OpenWSN. The latest builds of both the OpenPi and the OpenVM can always be found the OpenWSN CI servers.

IV. OpenMote: an open source prototyping platform for IIoT

OpenMote Technologies is a start-up company that provides a hardware ecosystem for the IIoT that is supported by the main open source stacks, i.e., OpenWSN, Contiki and RiOT.

The core of the OpenMote ecosystem [9] is the OpenMote-CC2538 board, depicted in Fig. 2, which is based on the Texas Instruments CC2538 SoC (System-On-Chip) and has a XBee-compatible form factor. The CC2538 features a 32-bit ARM Cortex-M3 micro-controller, with 32 kB of RAM memory and 512 kB of flash memory, and a radio transceiver compliant with the IEEE802.15.4 standard. The radio transceiver has a -97 dBm sensitivity level and a transmit power up to +7 dBm; the current consumption is 20 mA in receive mode and transmit current is 24 mA at 0 dBm. Last but not least, the

---

[3] https://openwsn.atlassian.net/wiki/display/OW/Hardware

[4] http://openwsn-builder.paris.inria.fr/

CC2538 includes an advanced security acceleration module, supporting AES-128/256, SHA2, ECC-128/256 and RSA. In addition to being more powerful than MSP430-based devices, a main advantage of the CC2538 architecture is that the radio registers and transmit/receive packet buffers are directly mapped to the processor memory. This dramatically reduces the delay for reading and writing packets from/to the radio compared to architectures based on the SPI bus, allowing to further reduce the energy consumption.

The OpenMote ecosystem also includes two companion boards for the OpenMote-CC2538, the OpenBase and the OpenUSB, which aim to facilitate prototyping. On the one hand, the OpenBase provides Ethernet connectivity (IEEE802.3, 10BASE-T) using the Microchip ENC28J60, and in-circuit programming and debug interfaces based on a JTAG interface (IEEE1149.7, cJTAG). On the other hand, the OpenUSB provides digital sensors, additional wireless connectivity and advanced programming interfaces. The OpenUSB has a TelosB form-factor and can be seamlessly powered via the Type-A USB connector or from 2xAA batteries. Regarding sensors, it features a temperature and relative humidity (Sensirion SHT21), light (Maxim MAX44009) and 3-axis acceleration (Analog Devices ADXL346) connected via the I2C (Inter-Integrated Circuit) bus. Regarding wireless connectivity, it supports the IEEE802.15.4g standard in the Sub-GHz bands using the Texas Instruments CC1200 radio transceiver. Finally, the OpenUSB has a current consumption measurement circuit based on an high-side instrumentation amplifier (Texas Instruments INA216), and supports automatic code boot-loading through the serial port, which greatly simplifies programming in large-scale test-bed scenarios.

V. OPENWSN & OPENMOTE DEMO

The demo showcases an IIoT network running the OpenWSN stack with OpenMote hardware as depicted in Fig. 1. The network consists of various OpenMote-CC2538 boards powered by OpenUSB boards. A number of these devices are powered using batteries and act as leaf nodes, whereas one device is connected to a computer running OpenVisualizer that acts as the network coordinator and bridge to the Internet.

As part of the demo, each leaf device periodically reports its sensor information to the Internet using CoAP-over-IPv6 data. Such information is stored in a database, which is then queried and displayed from the remote client. In addition, the remote client application running on the computer can interact with leaf devices individually by sending CoAP-over-IPv6 requests, i.e., switch LEDs on/off.

VI. CONCLUSION AND FUTURE WORK

The OpenWSN and OpenMote projects provide the tools required to rapidly prototype Internet-enabled sense and control applications and realize the IIoT vision. The OpenWSN project provides an open-source protocol stack that implements the IEEE low-power wireless technologies and the IETF Internet standards to enable robust end-to-end communications, whereas OpenMote provides an open-source hardware platform specifically tailored to the IIoT requirements.

As future work, communications in the sub-GHz bands based on IEEE802.15.4g standard will be integrated in the OpenWSN stack to support wide area deployments using the Texas Instruments CC1200 radio transceiver on the OpenUSB board. The integration of IEEE802.15.4-2015 with IEEE802.15.4g opens up new research and development opportunities, since multiple TSCH schedules operating in different bands will need to coexist in a single device.

REFERENCES

[1] N. Kushalnagar, G. Montenegro, and C. P. P. Schumacher, *IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs): Overview, Assumptions, Problem Statement, and Goals*, IETF Std. RFC4919, August 2007.

[2] J. Hui and P. Thubert, *Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks*, IETF Std. RFC6282, September 2011.

[3] 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*, IETF Std. RFC6550, March 2012.

[4] Z. Shelby, K. Hartke, and C. Bormann, *The Constrained Application Protocol (CoAP)*, IETF Std. RFC7252, June 2014.

[5] *802.15.4e-2015: IEEE Standard for Local and metropolitan area networks–Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs) Amendment 1: MAC sublayer*, IEEE Std., October 2015.

[6] T. Watteyne, A. Mehta, and K. Pister, "Reliability Through Frequency Diversity: Why Channel Hopping Makes Sense," in *International Symposium on Performance Evaluation of Wireless Ad Hoc, Sensor, and Ubiquitous Networks (ACM PE-WASUN)*. ACM, 26-30 October 2009.

[7] P. Thubert, T. Watteyne, M. R. Palattella, X. Vilajosana, and Q. Wang, "IETF 6TSCH: Combining IPv6 Connectivity with Industrial Performance," in *International Workshop on Extending Seamlessly to the Internet of Things (esIoT)*, Taiwan, 3-5 July 2013.

[8] T. Watteyne, X. Vilajosana, B. Kerkez, F. Chraim, K. Weekly, Q. Wang, S. Glaser, and K. Pister, "OpenWSN: A Standards-Based Low-Power Wireless Development Environment," *Transactions on Emerging Telecommunications Technologies*, vol. 23, no. 5, pp. 480–493, August 2012.

[9] X. Vilajosana, P. Tuset, T. Watteyne, and K. Pister, "OpenMote: Open-Source Prototyping Platform for the Industrial IoT)," in *7th International Conference, AdHocHets 2015*, vol. 155, Springer. San Remo, Italy: Springer, 12 Novermber 2015 2015.