Krzysztof PIOTROWSKI[1], Igor KOROPIECKI[1], Przemysław ZIELONY[1], Piotr POWROŹNIK[2], Robert SZULIM[2]
[1]IHP – Leibniz-Institut für innovative Mikroelektronik, Frankfurt (Oder), Germany
[2]Uniwersytet Zielonogórski
Instytut Metrologii Elektroniki i Informatyki

MOBILE APPLICATION TO SUPPORT MAINTENANCE OF WIRELESS SENSOR NETWORKS

This work describes the progress towards creating a tool realized as an application for mobile devices to maintain sensor nodes installed in the monitored environment using Bluetooth Low Energy. The application is generic, so it supports any set of monitored data and maintenance functions, as long as the node application includes the basic functionality required for the maintenance task.

APLIKACJA MOBILNA WSPIERAJĄCA ZADANIE UTRZYMANIA BEZPRZEWODOWEJ SIECI SENSORÓW

Praca ta opisuje postępy w tworzeniu narzędzia realizowanego jako aplikacja dla urządzeń mobilnych do obsługi węzłów sieci sensorów zainstalowanych w monitorowanym środowisku z wykorzystaniem technologii Bluetooth Low Energy. Aplikacja ma charakter ogólny, a więc obsługuje dowolny zestaw monitorowanych danych i funkcji konserwacyjnych, o ile aplikacja węzła zawiera podstawowe funkcje wymagane do realizacji zadania utrzymania.

1. MOTIVATION

The nodes of a sensor network are deployed in the environment they monitor and it is usually not easy to maintain them after the deployment. Using the same wireless channel for transmitting the measurements and maintenance data may affect the basic functionality of the nodes and, e.g., cause data losses. Additionally, the data channel is usually limited in throughput, so it is not possible to transmit larger data blocks without involving the maintained node a lot. This work presents the approach to use two different channels for measurement transmission and for maintenance. In our approach, the second functionality uses the Bluetooth technology, so it is possible to employ mobile devices for in-situ node maintenance without the need to equip them with additional communication module. The mobile device, equipped with the maintenance application, allows configuring the node and supports live data presentation without interfering much with its basic communication related to the monitoring functionality. The presented approach is being developed within the INTERREG project SmartRiver [1].

There are many approaches for cloud-based tools for management and monitoring of wireless sensor networks (WSN) [2]. These tools support many features needed for the different steps of the life cycle of a WSN. This support often starts with the design phase and includes deployment, maintenance of the network and processing and presentation of the collected data [3]. However, in many cases the cloud-based solutions use the same communication channel for both, collecting the data and the maintenance. This may be acceptable if the maintenance is limited to a simple monitoring of the basic functions of the nodes and does not affect the communication bandwidth, but becomes a problem if larger amounts of data need to be transmitted, e.g., to upgrade the firmware of the nodes. In these cases, either the maintenance process is stretched in time or another solution that involved a different communication channel is needed.

2. PROPOSED APPROACH

Most of the hardware platforms for wireless sensor networks support only a single communication module. This is mainly due to their most important feature, i.e., the constrained energy. However, there are also sensor nodes equipped with more than one communication module. This design choice can be driven by different reasons. The IHP node [4] is equipped with three radio modules and the reason behind that was to support communication redundancy and improve reliability. This platform was further evolving [5] and its variants were equipped with different communication modules, supporting different radio bands and by that using their features, like better signal propagation in the lower sub-GHz bands or higher throughput in the 2.4 GHz band. However, the used communication protocols were always proprietary, thus the communication between the nodes and a user device required an attachment to the (mobile) user device. So, even if the multiple communication modules allowed the use of one of them for maintenance purposes, without interrupting the basic inter-node communication in the sensor network, the fact that the smartphone or tablet needs a dongle attached, made it less comfortable. Anyway, we used that approach with a MS Windows operating system based tablet in the Sens4U project [6].

With the recent changes in the microcontroller architectures for wireless sensor networks and the vast availability of integrated communication modules based on standards, also these known from more powerful devices, like Bluetooth, direct communication between sensor nodes and mobile user devices became possible. The newest development at the IHP in the area of wireless sensor network, the MARS Node lite (see Fig. 1), uses the CC1352R System on Chip (SoC) solution that combines the sub-GHz radio module and a 2.4 GHz module, while the latter is capable of implementing the Bluetooth stack.

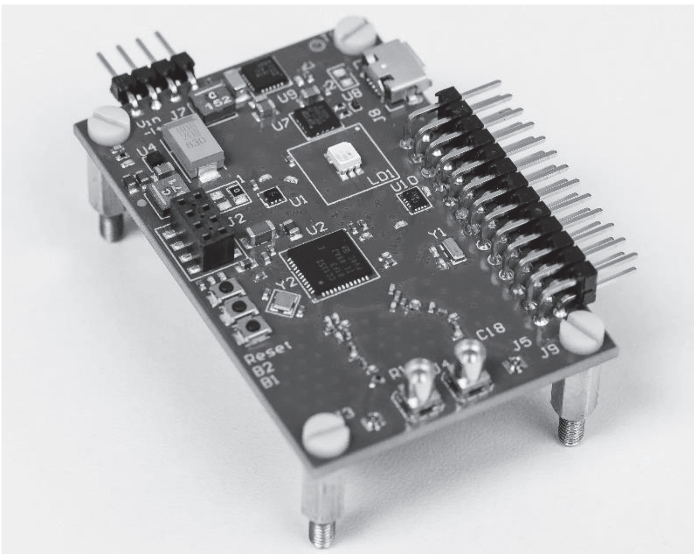

Fig. 1. The newest IHP sensor node hardware platform – MARS Node lite
Rys. 1. Najnowsza platforma sprzętowa IHP dla sieci sensorów – MARS Node lite

3.2 Maintenance functionality

The MARS Node lite is the base platform for the maintenance application this work presents. The nodes need a basic software module for the maintenance functionality available and once it is included in the node firmware, then every sensor node from the sensor network is accessible using the mobile device application. The following functions can be realized on the mobile device:

- Authorized connection to the sensor node from a smartphone or tablet,
- Uploading a new firmware using the Over the Air Download (OAD),
- Reading the measurement variables from the node,
- Reading and writing the configuration variables to the node,
- Presentation of the node state based on the values of its variables,
- Monitoring the node's functioning, by detecting defined events and specifying reactions (watchdog functionality).

The list of variables that are available on a specific node depends on the node's firmware and the mobile application needs to be generic to support any node firmware configuration. It is important that the node firmware was implemented using the maintenance library and that the data on the node is linked in the firmware with the variables. The measurements include the environment monitoring values, but also values that cover the node's own state, e.g., voltages or energy consumption. Variables that store the measurements are read-only variables, so the mobile application can only read these. Usually, measurement variables contain historic values, creating a time series that is available to the mobile application. On the other hand, the behavior of the node can be influenced by changing the values stored in the configuration variables, so these can be used to configure the node. Configuration variables can be both read and written, so their current state can be presented in the mobile application as well as their values can be updated. It is also important to mention that the connection between the mobile application and the node can expose a different level of privileges, i.e., depending on the user class not all variables may be accessible. The access control is realized using digital certificates that define the access level.

4.2 The communication protocol

In order to exchange the data between the node firmware and the maintenance application a simple data exchange protocol was defined. This protocol consists of commands and responses and in its current state the commands are always issued from the mobile application and the node replies.

The CONNECT command takes as parameter the *credentials* of the user. The reply of the node is the CHALLENGE reply containing the *challenge* value as parameter. In order to accomplish the connection the mobile application has to reply with the RESPONSE containing the *response* value calculated for the *challenge*. If the *response* is correct, the node replies with CONNECTED, and otherwise, with the ERROR reply. The positive outcome of the step starts the session between the two endpoints. To close the connection the mobile application issues a DISCONNECT request and the node replies with the DISCONNECTED reply.

The establishment of the connection is the only state-full part of the communication. There is a defined timeout for the mobile application to reply for the challenge. Once it runs out the state is not stored anymore and the node is not notifying the application about that.

In order to obtain the list of variables available on the node the mobile application issues the LIST request. The node replies with the VARLIST reply that holds a list of structures, one for each variable on the node, stating the *name* of the variable, its *type* (measurement or configuration), the *number of available values*, as well as the *first and last timestamp* for the values.

Knowing the names and the types of the variables, the mobile application can access these. In order to read the values of a chosen variable the mobile application issues the GET request, with the *name* of the variable and the *target timestamp range*, as parameters. The node replies with VALUELIST with a list of structures, each containing the *value* and *timestamp* representing the stored data.

On the other hand, the configuration variables can be written from the mobile application with the SET request that takes the *name* of the variable and the *value* to be written, as parameters. The node replies with the SETACK reply containing the *result* of the operation.

Additionally, in order to clear the storage on the node the mobile application can issue the DELETE request. The request takes the *name* of the variable and the *timestamp range* for data to be deleted as parameters. This operation never affects the most recent value. The node replies with the DELETEACK reply containing the *result* of the operation.

3. FURTHER STEPS

The partners in the SmartRiver project are currently developing this application. It is also a part of common teaching with the students at the University of Zielona Gora. Once we have the first version of the application we will evaluate it and prepare a list of desired extensions. In parallel, we also develop the node library to support the firmware development, i.e., to simplify the integration of the maintenance middleware and the basic functionality required on the nodes due to their monitoring tasks in the wireless sensor network. The goal is to use this solution to monitor the nodes deployed in the project. Such an interaction can simplify the evaluation of nodes' parameters and fine-tuning.

ACKNOWLEDGMENTS

This work was supported by the European Regional Development Fund within the BB-PL INTERREG V A 2014-2020 Programme, "reducing barriers – using the common strengths", project SmartRiver, grant number 85029892 and by the German government under grant 01LC1903M (AMMOD project). The funding institutions had no role in the design of the study, the collection, analyses, or interpretation of data, in the writing of the manuscript, or in the decision to publish the results.

REFERENCES

1. Michta E., Szulim R., Sojka-Piotrowska A., Piotrowski K.: *IoT based Flood Embankments Monitoring System*, Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments 2017, SPIE Proceedings, 2017.
2. Ray P.P.: *A survey of IoT cloud platforms*. Future Computing and Informatics Journal, Volume 1, Issues 1–2, pp. 35-46, ISSN 2314-7288, 2016.
3. Particle Inc. IoT Command Center, Product Web Page, https://www.particle.io/iot-command-center/, last viewed 26.04.2020.
4. Piotrowski K., Sojka-Piotrowska A.: *IHP Node – The Experimental Platform for Wireless Sensor Networks and Internet of Things*, in Proc. 11th Scientific Conference on Measurement Systems in Research and in Industry, pp. 121-124, 2016.
5. Piotrowski K., Sojka-Piotrowska A., Stamenkovic Z., Kraemer R.: *IHP Node Platform as a Base for Precision Farming and Remote Diagnosis in Agriculture*, in Proc. 24th Telecommunications Forum (TELFOR 2016), 2016.
6. Piotrowski K., Peter St.: *Sens4U: Wireless Sensor Network Applications for Environment Monitoring Made Easy*, in Proc. 4th International Workshop on Software Engineering for Sensor Network Applications (SESENA 2013), 2013.
7. Texas Instruments Inc.: CC1352R – SimpleLink™ 32-bit Arm Cortex-M4F multiprotocol Sub 1 GHz and 2.4 GHz wireless MCU, Product Web Page, www.ti.com/product/CC1352R, last viewed 26.04.2020.