OPEN ACCESS

International Electronic Conference on Sensors and Applications
1 - 16 June 2014

www.mdpi.com/journal/sensors

Conference Proceedings Paper – Sensors and Applications

Pogo-Pin-JTAG-Programmer-Box: A Low-Cost JTAG Programmer Interface for the Wireless Embedded Zolertia-Z1 Platform

Laurent Segers [1,*], Ruben Smeets [3], Matthias Carlier [2], An Braeken [1], Abdellah Touhafi [1,2], Kris Steenhaut [2], Nele Mentens [3]

[1] Department of Industrial Sciences and Technology (INDI), Vrije Universiteit Brussel, Pleinlaan 2, Elsene 1050, Belgium; E-Mails: an.braeken@vub.ac.be (A.B.); abdellah.touhafi@vub.ac.be (A.T.)

[2] Department of Electronics and Informatics (ETRO), Vrije Universiteit Brussel, Pleinlaan 2, Elsene 1050, Belgium: matcarli@vub.ac.be (M.C.); ksteenha@etro.vub.ac.be (K.S.)

[3] Technologiecluster Elektrotechniek (ESAT), Campus Diepenbeek (@ KHLim), Wetenschapspark 21,3590 Diepenbeek: ruben.smeets@kuleuven.be (R.S.); nele.mentens@esat.kuleuven.be (N.M.)

* Author to whom correspondence should be addressed; E-Mail: laurent.segers@vub.ac.be (L.S.); Tel.: +3-225-591-528.

Published: 1 June 2014

Abstract: In this paper, we propose a simple, reusable and non-intrusive approach for programming the Zolertia-Z1 mote through the JTAG-connection. Programming the Zolertia-Z1 mote is generally achieved by using the Contiki Boot Script Loader and a USB cable. However, complications arise when building programs which exceed the 64kB-boundary of program memory. Special care must then be taken in order to fully utilize the 92kB of available memory. At one hand, a 20-bit compiler is needed to support the complete available memory. At the other hand, a JTAG connection must be used in order to flash the program onto the MSP430-microcontroller of the Zolertia-Z1 platform. Solutions exist in the aim of facilitating the JTAG-programming. However, these solutions are generally not reliable, need an intrusive approach or are prohibitively expensive. Therefore, we propose a non-intrusive, reusable and reliable solution based on pogo pins. The Pogo-Pin JTAG Programmer Box consists out of an adapter board and the box itself. The adapter board can be used separately from the box for one-time flashing. Using both at the same time ensures reliable consecutive reprogramming of the same Zolertia-Z1 mote.

Keywords: Pogo-Pin; JTAG; Zolertia-Z1; 6LoWPAN; Sensor Networks; Contiki

---

1. Introduction

Wireless sensor networks are playing a more important role in our society. Applications of wireless sensor networks include home automation [3,4], environmental monitoring and health care [6,7]. Several wireless protocols have been developed in the aim of facilitating the implementation of applications [1,3,4]. In our project, "6LoWPAN: Towards Zero Configuration", we develop a home automation system which uses IPv6 for low power wireless private area networks (6LoWPAN). The goal of the project is twofold: at one hand we develop a system that reduces human intervention during the setup and, at the other hand; we also implement suitable security mechanisms in order to prevent intrusion by undesired persons and systems. Multiple security issues are known in the context of wireless sensor networks [5]. However, our main security focus is to prevent that intruders are able to access sent data, especially modifying data and taking undesired actions (i.e. toggling a light bulb).

The development platform chosen in our project is the Z1 mote delivered by Zolertia [11], which supports the Contiki operating system with a built-in 6LoWPAN software stack [2]. The open-source MSP430-GCC compiler is used to compile the applications for this platform. Until release 4.6.3, the compiler only supported 16-bit addressing capabilities (i.e. 64kB addressable memory). The latest release (i.e. 4.7) of the compiler makes it possible to fully address the Zolertia Z1's memory space. However, the current bootstrap script loader (BSL) of Contiki does not allow to flash the mote with programs exceeding the 64kB boundary. To aleviate this shortcoming, the Joint Test Action Group (JTAG) connection can be used to fully program the mote. Due to the small size of the onboard JTAG connector, programming the Z1 mote can be a fastidious task. To augment the programming efficiency, we build an easy to use programmer adapter system: the Pogo Pin JTAG Programmer Box.

The remainder of this paper is divided into following sections: in Section 2 we introduce the application together with the Contiki operating system and the Zolertia Z1 mote. In Section 3 we discuss the different existing methods for reflashing the Zolertia Z1 mote. In Section 4 we introduce our low-cost Pogo Pin JTAG Programmer Box and in Section 5 we discuss the concluding notes.

2. The Contiki Operating System and the Zolertia Z1 Platform

Our project consists out of building a home automation system with a wireless sensor network based on the 6LoWPAN protocol. The main goal of our project is to reduce the needed human intervention while setting up the system. Although the 6LoWPAN standard defines security mechanisms, these have not been fully implemented yet. Therefore, the second focus of our project lies in the protection of the system from external intruders. The platform chosen on which our project is developed, is the Z1 mote from Zolertia which runs the Contiki operating system.

The Zolertia Z1 mote contains a MSP430F2617 micro-controller [9] from Texas Instruments running at 16 MHz. The platform provides digital input-output (IO), digital communication, interruptable pins and a USB connection for universal asynchronous receiver/transmitter (UART) communication. The board also offers RF connectivity through an on-board ceramic antenna or through an attached external antenna (predefined at manufacturing). The MSP430F2617 also has a total available memory of 92kB. Aside of these capabilities, the board also provides a small JTAG connection [9].

The Z1 mote runs natively the Contiki operating System [2]. This operating system provides the ability of event driven proto-threads, drivers for the underlying micro-controller (i.e. I2C, interrupt handler for a button, etc.), and facilities for the 6LoWPAN connectivity. The complete application (i.e. the user application and the Contiki operating system) is compiled using the open source MSP430-gcc compiler [8]. The MSP430 micro-controller family is based on a 16-bit processor architecture, which reduces the total addressable memory space to 64kB. However, some MSP430 micro-controllers (i.e. MSP430F2617) are equipped with more memory than addressable by 16 bits and therefore support the extended instruction set with 20-bit addressing capabilities. This 20-bit addressing ads complexity for the compiler and until recently the MSP430-gcc open source compiler did not support this extended feature. This lack of support also had repercussions in the way developers of Contiki implemented the operating system.

Zolertia offers several kinds of packets to customers. We chose the Gold packet, which includes 10 Z1 motes boxed in a white enclosure with an external antenna. The current setup of the nodes makes it more difficult to add additional headers on the board (i.e. the JTAG header, a header on the South-port of the board).

Programming the nodes can be achieved by using two main methods. These methods are detailed in following sections.

3. Flashing the the Zolertia Z1

Several approaches exist in order to program the Zolertia Z1 (MSP430 micro-controller) motes. These can be categorized in two main methods: using the on-board USB connection in conjunction with a boot script loader, and the JTAG connection. The current methods are explained below.

3.1. USB-programming

The first and probably most used method to program the Zolertia Z1 mote utilizes a serial over USB connection in combination with a bootstrap script loader (BSL). The Z1 mote has a USB-to-serial converter which allows to program the mote using the on-board USB connection. The bootstrap script loader is a small add-on in the micro-controller which allows to reprogram the controller with a predetermined data sequence. When the predetermined data sequence is satisfied, the mote is reflashed. The bootstrap script loader of Contiki loads the compiled application into the Z1 mote by invoking the BSL on the microcontroller. This script was primarily targeted for programs smaller than 64kB. Programs exceeding this boundary are not flashed because the conditions of the predetermined data sequence for the bootstrap loader are not met.

3.2. JTAG programmer

The second method for reflashing the micro-controller is achieved by using the available JTAG connection. Reprogramming the Zolertia Z1 mote through JTAG is done using the small dual row header of eight connection points, with a spacing of 50 mills. Because of this small spacing (i.e. compared to a standard 100 mill spacing), Zolertia offers 2 possibilities to program the mote using JTAG: a small adapter board and the extend version of the Z1 platform.

3.2.1. Zolertia's Adapter Board

The Zolertia's adapter board is composed of a connector, a small header and a power jumper. The small header on the board connects the mote to the board, while the connector is connected to the MSP-FET430UIF JTAG-programmer. The power jumper allows the programmer to choose whether the mote is fed by the JTAG programmer or by the main supply of the mote itself. This adapter board comes up with some packs delivered by Zolertia. To connect the mote to the adapter board, one can solder the board directly onto the mote, or press it firmly against the board to be certain all electrical junctions are made. The latter method makes this system unreliable and a cumbersome task if one wants to program several nodes.

3.2.2. Zolertia’s Extended Z1 Platform

A second more recent approach consists by using the Z1 Starter Platform from Zolertia. This mote facilitates the IO capabilities by bringing all connections to the external casing. An additional larger JTAG MSP-FET430UIF-compatible connection has also been added to the design. Although a lot of features have been added, the downside is the additional price per node, which is € 253. Compared to the Gold pack of Zolertia, which contains 10 nodes for € 999, the additional price per nodes is around € 153, which could be a barrier for medium to large scale test setups.

4. Pogo Pin Based JTAG Programmer

To alleviate the shortcomings of the previously cited methods, we developed our pogo-pin based JTAG adapter system. Pogo-pins are generally used within testing facilities, where they are mounted onto a bed and test the electrical connections of many printed circuit boards. The latter makes these pins suitable for a reusable JTAG adapter board design. Our system consists out of an adapter board, which can be complemented by a housing box.

4.1. Pogo Pin Adaptor Board

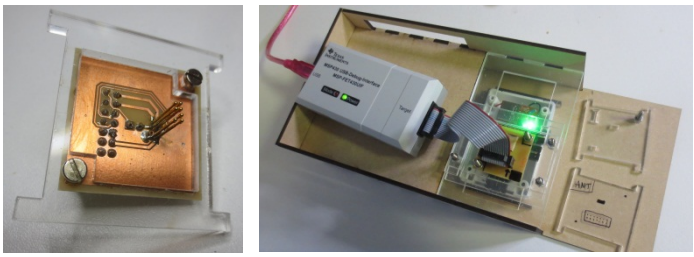
The pogo-pin adapter board is mainly based on the design of the adapter board delivered by Zolertia. However, additional mounting holes have been added to the design and the regular header has been replaced by pogo-pins [10]. The diameter of the pins are slightly larger than the hole-diameters of the Zolertia Z1 JTAG connection and have total expansion length of 2,65 mm. The internal spring ensures that the head of the pin is pushed towards the connection point. By such, the pogo-pins allow the programmer to press the board towards the mote and being sure all connections are made. The mounting holes on the adapter board allow one to mount the printed circuit board onto a 3 mm Perspex plate. The latter setup is useful in combination with the housing box (see Figure 1).

4.2. Housing Box

The purpose of the housing box (Figure 1) is to hold the adapter board connected to the Zolertia Z1 mote during consecutive programmings (i.e. when testing new code). By doing so, one can easily reprogram the Zolertia Z1 mote without readjusting the adapter board onto the mote (which would be

the case with the adapter board described in section 3.2.1.). A Perspex window of 3mm thick allows the programmer to see the behavior of the LEDs. The wooden part of the housing box is made out of 3mm medium density fiberboard (MDF). The platform's height of the housing box matches the necessary height for the pogo-pins in such way that the pins are able to move in the specified spring tolerance of 2,65mm. An additional Perspex window allows one to carefully place the Zolertia Z1 mote into the housing box before mounting the pogo-pin adapter board.

Figure 1. Adapter board (**left**) and the complete pogo pin programmer with the housing for the Z1 mote (**right**). The adapter board is fixed by a sliding mechanism made out of Perspex.



4.3. Manufacturing cost

Both the adapter board and the housing box have been manufactured with the equipment of the FablabXL of Brussels. A laser cutter has been used to cut the materials into the desired shapes, which considerably reduced manufacturing costs. A small PCB has been edged to hold the pogo-pins and to ensure electrical connection between the MSP-FET430UIF JTAG-programmer and the Zolertia Z1 mote. The total manufacturing costs are shown in **Table 1**.

Table 1. Material costs for the production of the complete setup.

Material	Cost
MDF	€0.5
Perspex	€1
PCB	€1
Pogo-pins (7)	€1,23 · 7 = €8,61
Total	€11,11

These costs include the materials used and the cost for manufacturing the complete setup. As one can see, the total cost is far below the cost proposed for reliable solutions. The adapter board is also more reliable in use then the one provided by Zolertia with some packs. In addition, the housing box allows the programmer to free up his hands while programming.

5. Conclusions

In this paper we presented a low-cost alternative to current available methods for programming the Zolertia Z1 mote for programs exceeding the 64kB boundary. The setup of our system is twofold. At one hand one can find the adapter board which makes it possible to make reliable connections between the Zolertia Z1 mote and the MSP-FET430UIF JTAG-programmer. At the other hand, a housing box allows the programmer to program a mote multiple times without readjusting the position of the

adapter board. The use of pogo-pins allows one to reuse the system several times without the need of an intrusive approach.

Acknowledgments

This research has been made possible thanks to a Tetra grant of the Flanders agency for Innovation by Science and Technology. The authors would like to thank the FablabXL of Brussels, the fabrication laboratory where all components have been manufactured. The authors would also like to acknowledge the contribution of the COST Action IC1303 - AAPELE, Architectures, Algorithms and Platforms for Enhanced Living Environments.

Conflicts of Interest

The authors declare no conflict of interest.

References and Notes

1. Dunkels, A.; "Contiki: Bringing IP to Sensor Networks", *Ercim news*, 2009, Vol. 76, pp. 59-60.
2. Dunkels, A.; Gronvall, B. ; Voigt, T. "Contiki - a lightweight and flexible operating system for tiny networked sensors", *29th Annual IEEE International Conference on Local Computer Networks*, Tampa, USA, 16-18 November 2004. pp. 455-462.
3. Gill, K.; Yang, S.-H.; Yao, F.; Lu, X. "A ZigBee-based home automation system" *IEEE Transactions on Consumer Electronics, vol. 55*, 2009, 2, pp. 422–430.
4. Sriskanthan, N.; Tan, F.; Karande, A. "Bluetooth based home automation system" *Microprocessors and Microsystems, vol. 26*, 2002, 6, pp. 281–289.
5. Le, A.; Loo, J.; Lasebae, A.; Aiash, M.; Luo, Y. *"6LoWPAN: a study on QoS security threats and countermeasures using intrusion detection system approach", Int. J. Comm, vol. 25*, 2012, 9, pp. 1189-1212.
6. Ko, J.; Lu, C.; Srivastava, M. B.; Stankovic, J.A.; Terzis, A.; Welsh, M. *"Wireless Sensor Networks for Healthcare", Proceedings of The IEEE, vol. 98*, 2010, 11, pp. 1947-1960.
7. Dominguez, F. Design and Realization of a Low-Cost, Location-aware Sensor Network for Noise Pollution Monitoring, Doctoral thesis, Vrije Universiteit Brussel, Brussels, 01 April 2014.
8. MSP430-GCC Support Page: Available online: http://sourceforge.net/apps/mediawiki/mspgcc/index.php?title=MSPGCC_Wiki (accessed on 21 February 2014).
9. Zolertia Z1 Datasheet: Available online: http://zolertia.sourceforge.net/wiki/images/e/e8/Z1_RevC_Datasheet.pdf (accessed on 28 February 2014).
10. Pogo-Pin Datasheet. Available online: http://www.farnell.com/datasheets/1680273.pdf (accessed on 6 March 2014).
11. Website of Zolertia. Available online: http://zolertia.com (accessed on 6 March 2014).

© 2014 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution license (http://creativecommons.org/licenses/by/3.0/).