Egs: A Cortex M3-based Mote Platform

JeongGil Ko[†] Qiang Wang[‡] Thomas Schmid[*] Wanja Hofer[⋆] Prabal Dutta[*] Andreas Terzis[†]

[†] Department of Computer Science, Johns Hopkins University

[‡] Department of Control Science and Engineering, Harbin Institute of Technology

[*] Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor

[⋆] Department of Computer Science, Friedrich−Alexander University Erlangen−Nuremberg

***Abstract*—We introduce the *Egs* mote platform based on the Cortex M3 microcontroller that focuses on medical sensing applications. Egs uses an Atmel SAM3U microcontroller that runs up to 96 MHz and has up to 52 KB of RAM and 256 KB of Flash. Egs combines this microcontroller with two radios (802.15.4 and Bluetooth), external flash, on board sensors, and a LCD touchscreen to enable a rich set of wireless sensing applications.**

I. Introduction

Wireless sensor networks (WSNs) consist of low-power, embedded computing devices, known as *motes*, that use sensors to collect measurements from the physical world and its inhabitants. Low-power sensors combined with wireless mote platforms can be easily deployed to various environments and thus are widely used in many different real-life applications from environmental [12] and structural health monitoring [14] to medical sensing applications [8].

However, while existing mote platforms, such as the TelosB [11] and MICAz [3] have the low power consumption necessary for long-term deployments, their computational and memory resources are highly restricted. For example, the widely used MSP430x1xx microcontroller (MCU) runs at a speed of 8MHz and the TelosB mote (which uses the MSP430x1xx) has only 10KB of RAM and 48KB of ROM. Such lack of resources limits the types of applications that these mote platforms can support. In fact, the original target applications of these platforms are far from how they are trying to be used in some of the recent deployments. For example, many applications today, including medical sensing applications, can benefit from the use of a LCD screen to improve the user's experience. However, using LCDs require a large portion of the RAM, ROM and also more processing time. Furthermore, to increase the number of applications that these motes can operate in, developers try to implement complex compression or signal processing algorithms despite the resource constraints.

While many other new mote platforms have been developed since then (e.g., Iris [4], Epic [6], Shimmer [9], and BTNode [2]) most of these platforms are still based on similar MCUs as the one used on the TelosB described above. Some platforms including the iMote2 [5] provided developers with higher computational power (processing speed up to 416MHz with 256KB of RAM for the iMote2) at the expense of reduced energy efficiency and therefore could not be used in applications that require long-term and unattended deployments.

In this article we introduce a new mote platform that is designed with the goal of increasing computational power while maintaining low energy consumption. Specifically, we introduce the *Egs* mote platform that is based on the recently developed Cortex M3 processor [15]. The Cortex M3 is a 32-bit ARM processor based on the ARMv7-M architecture that is designed specifically for power and cost effective system performance in embedded applications. Egs uses the Atmel SAM3U2C [1] (which uses the ARM Cortex M3 as its core) and combines it with a Bluetooth and a Zigbee radio for wireless communications. Egs also has an on-board micro SD card slot along with a 2Gb NAND flash and a 2.4 inch touch screen LCD. The current focus of the Egs platform is medical sensing applications and for this reason it includes a UART port that connects to a pulse oximetry sensor board and a DB9 connector that connects to a finger tip pulse oximetry sensor.

This abstract introduces the hardware components of the Egs mote and the current status of software support in TinyOS 2.x. in detail. We also outline several potential applications which can benefit from Cortex M3-based platforms such as Egs.

II. The Egs Platform

A. Hardware Platform

The Egs platform is designed with the goal of providing wireless sensor network developers with a hardware platform that provides a sufficient amount of resources in terms of processing speed, memory size, and data storage. At the same time, to ensure that the new platform is applicable to long-term, unattended deployments, we keep energy efficiency as Egs' main design goal.

To achieve both computational power and energy efficiency, Egs is based on Atmel's SAM3U2C microcontroller [1]. The Atmel SAM3U2C uses the 32-bit ARM Cortex-M3 processor as its core, has a SRAM size of 36 KB, flash size of 128 KB and provides control for various peripheral controllers along with a memory protection unit (MPU). The SAM3U2C can operate at a core clock speed of up to 96 MHz with a current draw of 48 mA in active mode. The peripheral controllers provided by the SAM3U2C allow the integration of various external components. Specifically, the Egs platform includes a 2 Gb NAND flash and an external micro SD card slot to

Fig. 1. Picture of the Egs platform. The front side of the Egs platform includes a 2.4 inch LCD and two user buttons (left). The Atmel SAM3U2C microcontroller, TI CC2520 radio, Mitsumi Bluetooth module, 2 Gb NAND flash, and a micro SD card slot is located on the back side of the board (right). The pins that are located next to the micro SD card slot are used for the JTAG and can be detached at the end of the development phase.

make large amounts of data storage possible. Also, for wireless communications, the platform provides a TI CC2520 802.15.4 radio [17] and Mitsumi's class 2 Bluetooth module [10] both integrated with chip antennas. The front side of the device is covered with two user buttons and a 2.4 inch LCD touch screen with a resolution of 240 $\times$ 320 pixels. Finally, the device is currently powered with a mini-USB connection but will support a re-chargeable battery for mobile operations.

We plan to initially use Egs in our medical sensing application [8]. For this reason the current version of the the Egs platform includes UART pins that connect to the Nellcor pulse oximetry sensor board [13] and a DB9 pin connector for connecting the finger tip sensors as well.

B. Software Platform

We are currently adding support for the Egs platform to TinyOS 2.x. The current implementation supports various peripheral controllers including the USART, UART, ADC, I2C, SPI, and DMA controllers. We use these peripheral controllers to provide drivers for the external components described above. We note that most of our software development is not Egs specific but is general enough support all Atmel SAM3U-based microcontrollers (including more advanced MCUs such as the SAM3U4E).

III. Applications of Egs

With the increased computational power and low power consumption, we predict that Egs can improve the quality of existing wireless sensor network applications. Furthermore, we envision Egs as a driving force in enabling new applications for mote-class wireless sensor networks. Below, we outline several applications in which using a Cortex M3-based mote, such as Egs, can improve the quality of sensing and also the performance (i.e., energy efficiency, reliable end-to-end data delivery) of a wireless sensing system.

Improving Medical Sensing Applications. Medical sensing applications require many different sensors to collect data from various parts of a human body to make accurate medical decisions. However, due to the limited computational resources, existing medical sensing applications such as MEDiSN [8] can only support few sensors. One way to improve the sensing quality (or quantity) using the Egs platform is by enabling the Bluetooth radio to wirelessly collect sensor measurements from various sensors after forming a body area network (BAN). Moreover, given the platform's increased computational power, effective in-network processing (e.g., ECG compression, physiological signal pattern detection) can happen at the mote. Doing so reduces energy consumption by reducing the amount of data transmitted over the radio.

Increasing WSN Routing Capabilities. Recently, Schmid et al. proposed a novel network architecture for wireless sensor network deployments [16]. The work proposes the use of a multiple tier network architecture in which the sensors themselves have highly restricted resources and communicate to routers that have enough computational and energy resources to deal with the data aggregation and routing of the collected data (e.g., via mesh routing). Forming such a network architecture (specifically the high performance router node) is difficult to achieve with existing mote platforms due to their resource constraints. Even today, when a collection based WSN is formed, the central node (i.e., the root node in a tree network), where all the sensor measurements are aggregated, is often the bottleneck of the entire system's performance. Deploying a more powerful mote to reduce the overload at these critical points, can significantly improve the performance of a wireless sensor network deployment.

Connecting WSNs to the Internet. There is current interest to connect all networking objects, including miniature sized sensor motes, to the Internet [18] using technologies such as 6LoWPAN [7]. While providing each mote with an IPv6 stack has multiple advantages, the large memory and code size overhead of the address compression techniques may leave only a small amount of space for the user's application code to fit [16]. The additional resources that Cortex M3-based mote platforms such as Egs provide can be used to accommodate more complex user applications that previous motes cannot support.

References

[1] Atmel Corporation. AT91 ARM Cortex-M3 based Microcontrollers: SAM3U Specifications, 2009.

[2] Jan Beutel. Fast-prototyping using the btnode platform. In *DATE '06: Proceedings of the conference on Design, automation and test in Europe*, pages 977–982, 3001 Leuven, Belgium, Belgium, 2006. European Design and Automation Association.

[3] Crossbow Corporation. MICAz Specifications, 2004.

[4] Crossbow Corporation. Iris Specifications, 2007.

[5] Crossbow Inc. Imote2: High-Performance Wireless Sensor Network Node. Available at: http://www.xbow.com/Products/Product_pdf_files/Wireless_pdf/Imote2_Datasheet.pdf, 2007.

[6] Prabal Dutta, Jay Taneja, Jaein Jeong, Xiaofan Jiang, and David E. Culler. A building block approach to sensornet systems. In *Proceedings of the 6th International Conference on Embedded Networked Sensor Systems (SenSys)*, pages 267–280, November 2008.

[7] Jonathan Hui and David Culler. Ip is dead, long live ip for wireless sensor networks. In *SenSys*, 2008.

[8] JeongGil Ko, JongHyun Lim, Yin Chen, Razvan Musaloiu-E., Andreas Terzis, Gerald Masson, Tia Gao, Walt Destler, Leo Selavo, and Richard Dutton. MEDiSN: Medical Emergency Detection in Sensor Networks. *ACM Transactions on Embedded Computing Systems (TECS), Special Issue on Wireless Health Systems*, 2010.

[9] Konrad Lorincz, Benjamin Kuris, Steven M. Ayer, Shyamal Patel, Paolo Bonato, and Matt Welsh. Wearable wireless sensor network to assess clinical status in patients with neurological disorders. In *IPSN '07: Proceedings of the 6th international conference on Information processing in sensor networks*, pages 563–564, New York, NY, USA, 2007. ACM.

[10] Mitsumi Electric Co. WML-C46 Class 2 Bluetooth Module, 2006.

[11] MoteIV Corporation. Tmote Sky. Available at: http://www.sentilla.com/moteiv-endoflife.html.

[12] R. Musăloiu-E., A. Terzis, K. Szlavecz, A. Szalay, J. Cogan, and J. Gray. Life Under Your Feet: A Wireless Soil Ecology Sensor Network. In *Proceedings of the Third Workshop on Embedded Networked Sensors (EmNets 2006)*, May 2006.

[13] Nellcor Puritan Bennet Inc. OxiMax NELL-1: OEM Pulse Oximetry Module. Available at: http://www.nellcor.com/Serv/manuals.aspx?ID=291, 2006.

[14] Jeongyeup Paek, Krishna Chintalapudi, John Cafferey, Ramesh Govindan, and Sami Masri. A wireless sensor network for structural health monitoring: Performance and experience. In *Proceedins of the Second IEEE Workshop on Embedded Networked Sensors (EmNetS-II)*, May 2005.

[15] S. Sadasivan. An Introduction to the ARM Cortex-M3 Processor. 2006.

[16] Thomas Schmid, Roy Shea, Mani B. Srivastava, and Prabal Dutta. Disentangling wireless sensing from mesh networking. In *Proceedings of HotEmNets*, 2010.

[17] Texas Instruments. CC2520: Second generation 2.4 GHz IEEE 802.15.4 / ZigBee-ready RF Transceiver. Available at http://www.ti.com/lit/gpn/cc2520, 2007.

[18] T. Winter, P. Thubert, and RPL Author Team. Rpl: Ipv6 routing protocol for low power and lossy networks. Internet Draft, IETF, 2010.