ELSEVIER

CrossMark

Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 37 (2014) 189 – 195

Procedia Computer Science

5th International Conference on Emerging Ubiquitous Systems and Pervasive Networks (EUSPN-2014)

DZ50: Energy-Efficient Wireless Sensor Mote Platform for Low Data Rate Applications

Abdelraouf Ouadjaout*, Noureddine Lasla, Miloud Bagaa, Messaoud Doudou, Cherif Zizoua, Mohamed Amine Kafi, Abdleouahid Derhab, Djamel Djenouri, Nadjib Badache

CERIST Research Center, 05 rue des 3 Frères Aissou, Ben Aknoun, Algeria

Abstract

A low cost and energy efficient wireless sensor mote platform for low data rate monitoring applications is presented. The new platform, named DZ50, is based on the ATmega328P micro-controller and the RFM12b transceiver, which consume very low energy in low-power mode. Considerable energy saving can be achieved by reducing the power consumption during inactive (sleep) mode, notably in low data rate applications featured by long inactive periods. Without loss of generality, spot monitoring in a Smart Parking System (SPS) and soil moisture in a Precision Irrigation System (PIS) are selected as typical representative of low data rate applications. The performance of the new platform is investigated for typical scenarios of the selected applications and compared with that of MicaZ and TelosB. Energy measurements have been carried out for different network operation states and settings, where the results reveal that the proposed platform allows to multiply the battery lifetime up to *7* times compared to MicaZ and TelosB motes in 10*s* sampling period scenarios.

Peer-review under responsibility of the Program Chairs of EUSPN-2014 and ICTH 2014.

Keywords: Wireless Sensor Networks; Energy Consumption; Mote Platform; Low Rate Applications; TinyOS.

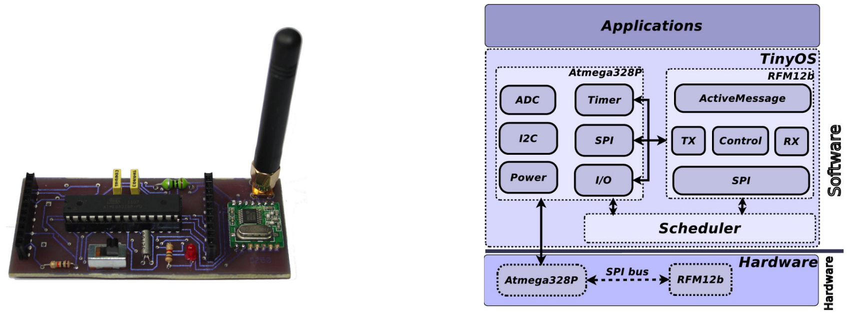
1. Introduction

The advances in wireless communications and embedded systems have enabled the development of *Wireless Sensor Networks* (WSN)[1]. These systems are composed of one or multiple base stations and a large number of miniature wireless sensor nodes. Each sensor node is equipped with a low energy supply, a small storage capacity and a short-range communication transceiver. Each sensor node can be equipped with one or more sensor devices that allow it to probe its surroundings and sense its environment. The sensed data are routed through multi-hops wireless communications to the base station for further analysis and use.

Environmental monitoring is a common application for WSN, where data are collected at specific rate and forwarded to a control station[2,3]. The most compelling purpose for these applications is the capability to design optimized platforms that extend the system lifetime at the order of years before battery/node replacement. The design of

* Corresponding author. Tel.: +213-21-912126; fax: +213-21-912126; email: aouadjaout@mail.cerist.dz

Peer-review under responsibility of the Program Chairs of EUSPN-2014 and ICTH 2014.
doi:10.1016/j.procs.2014.08.029



(a) Hardware architecture (b) Software architecture

Fig. 1. DZ50 architecture.

Table 1. DZ50, MicaZ and TelosB hardware specifications

	DZ50	*MicaZ*	*TelosB*
MCU	Atmega328P	Atmega128L	MSP430F1611
Architecture	Harvard 8 bits	Harvard 8 bits	Von Neumann 16 bits
Flash	32 KB	128 KB	48 KB
SRAM	2 KB	4 KB	10 KB
I/O pin count	23	53	48
Tranceiver	RFM12b	CC2420	
Frequency band	433, 868, 915 MHz	2.4 GHz	
Data rate	115.2 kbps	250 kbps	

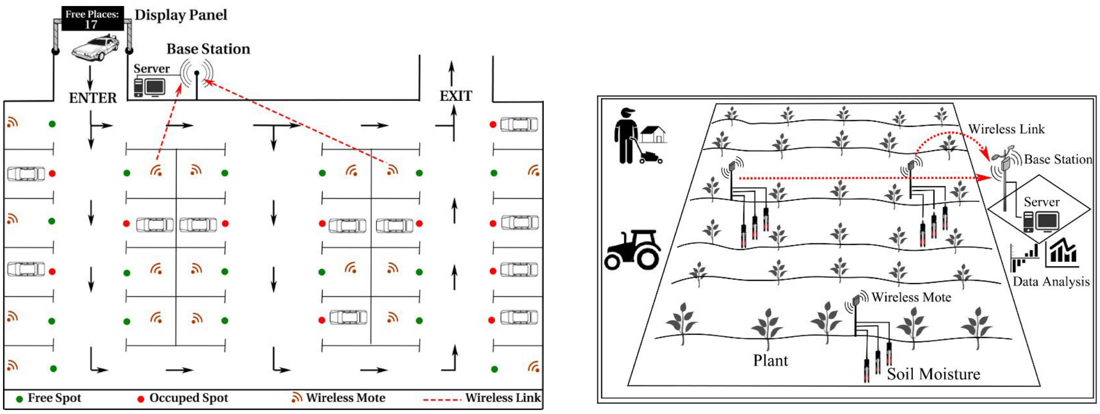
such systems is challenging and depends on the underlying software and hardware platform, as well as on the data load of the application. In applications where low data sampling rate is sufficient, sensor nodes pass most of the time in low-power mode and switch to active mode only when it is necessary. Timely switching to low-power mode during node inactivity allows significant energy saving and can appreciably extend the system lifetime. Low data rate applications are targeted herein, where a low-cost and energy-efficient wireless mote platform, named *DZ50*, is developed. The platform tends to extend the system lifetime to many years compared to well known wireless mote platforms, namely MicaZ[4] and TelosB[5].

The remainder of the paper is organized as follows. The software and hardware architecture of DZ50 is presented in section 2. In section 3, we provide two typical scenarios, a *Smart Parking System* (SPS) and a *Precision Irrigation System* (PIS), to show the efficiency of DZ50 mote. The empirical results are discussed in section 4. Finally, the paper is concluded in section 5.

2. Platform Architecture

The main design goal of DZ50 is to minimize energy consumption in low-power mode. For this purpose, DZ50 is composed of an ATmega328P micro-controller and a RFM12b radio transceiver, as depicted in Fig. 1(a). These components are characterized by their very low energy consumption: ATmega328P consumes less than $1\mu A$ in power-save mode, whereas RFM12b consumes less than $0.3\mu A$ in its standby mode. To allow the ATmega328P to enter the asynchronous power-save mode, an RTC $32Khz$ crystal is used. DZ50 is a modular platform, which extends its domain of applications. Moreover, DZ50 is designed as a self-contained battery-operating mote, which facilitates the deployment.

Despite the advantages cited above, DZ50 has some limitations related to its performance. In terms of memory, DZ50 has a lower Flash and SRAM capacity than MicaZ and TelosB. Moreover, the RFM12b transceiver in DZ50 allows only a data rate of 115.2 kbps, whereas CC2420 found in MicaZ and TelosB can reach 250 kbps. A summary of the hardware specification of the three platforms is given in Table 1. Nevertheless, these limitations do not represent a real hurdle in using DZ50 in low rate applications targeted in this work, but in fact they make it less eager in terms of energy consumtion.



(a) Floor layout of a *Smart Parking System*

(b) Soil moisture collection in a *Precision Irrigation System*

Fig. 2. Low data rate application scenarios.

We have ported all device drivers of DZ50 to TinyOS 2.x, which eases the programming of the platform and allows using many protocols already developed for TinyOS platform. As depicted in Fig. 1(*b*), we have developed six modules for Atmega328P driver part: *ADC*, *SPI*, *I2C*, *I/O*, *Timer* and *Power*. The four first components are used to establish communication between the micro-controller and external components. The *Timer* module manages the asynchronous hardware timer `Timer2`. The *Power* module encapsulates the access to the *Sleep Mode Control Register* (`SMCR`) to switch the mote in different modes (*i.e.*, active, sleep) to save the energy consumption.

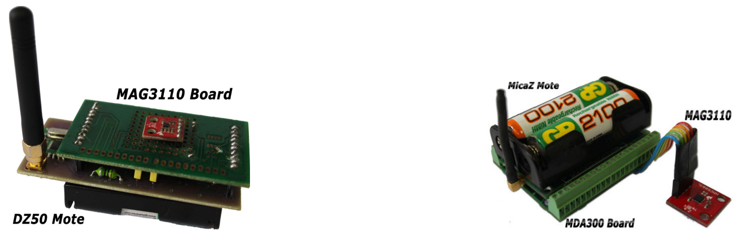
To simplify access to the functions of the RFM12b chip, we have developed five modules for its driver. The *Active Message* module allows the application to transmit and receive packets via an abstract interface. The *Control* module manages the different configurations of the chip, such as the frequency and the rate. The *Tx* and *Rx* components implement the low-level mechanisms to transmit and receive packets, respectively. The communication between the *Control*, *Tx* and *Rx* components, from one side, and the RFM12b hardware, from the other side, is established via an SPI serial communication. *Control*, *Tx* and *Rx* components send commands and receive events from the *SPI* component, which is located at RFM12b part. The latter communicates with RFM12b hardware via *SPI* component located in the micro-controller.

3. Illustrative System Scenarios

For illustration, a typical scenario of a *Smart Parking System* (SPS) and a *Precision Irrigation System* (PIS) have been considered in this work as representative of low rate (low data sampling) applications. These scenarios have been chosen to exemplify the application of the platform to a monitoring system, and to show its effectiveness in extending the system lifetime. Many WSN-based solutions have been proposed in the literature,[6,7,8,9], and system architecture of such solutions is out of the scope of this work. The focus here is to investigate the effectiveness of the proposed platform in this class of applications from the energy-consumption perspective. For the sake of simplicity, a star topology (single-hop network) is adopted in both applications where data samples are sent periodically to the central station for analysis. In the following, we give a brief description of each application.

3.1. Smart Parking System

Places in most of parking locations tend to be full, especially during the peak hours when drivers waste precious time looking for a free parking spot. The considered parking system periodically provides information concerning the number and locations of free spaces. Low data sampling rate is assumed, which is sufficient to detect and report the presence of vehicles. Referring to the layout in Fig. 2(*a*), sensor motes placed in different parking spots gather information about spots occupancy and report notifications to the base station. The base station on the other hand,



(a) (b)

Fig. 3. Hardware setup for the SPS application using the MAG3110 magnetometer with (a) DZ50 and (b) MicaZ + MDA300.

(a) (b)

Fig. 4. Hardware setup for the PIS application using the EC-5 sensor with (a) DZ50 and (b) TelosB.

located at the parking entrance, collects report messages from deployed motes and sends them to a central server via a serial connection. The gathered information can be analyzed and presented to drivers using display panels, and they can be used through the internet to provide customers with a variety of services in the context of internet of things.

3.2. Precision Irrigation System

The use of traditional irrigation techniques considerably reduces water preservation. More than %50 of water would be lost in the irrigation use[10]. Moreover, the use of traditional irrigation techniques have a negative effect on the environment, such as the salinization, intrusion of brackish water, ... *etc*. For this purpose, the use of an efficient irrigation control system becomes essential. An efficient irrigation control using a wireless sensor network is considered as the second target application. The sensors are deployed in the agriculture area, where every sensors is equipped with a set of soil moisture sensors as depicted in Fig. 2(*b*). The sensor nodes detect the amount of water in the soil via the soil moisture sondes. To give an accurate detection, the soil moisture sondes may be placed on different levels of the ground. Periodically, the sensor nodes measure the amount of water in the soil and then communicate the measured results to a control station (*i.e.*, the base station). The collected report messages can be used for automatic control of solenoid valves, or saved in a database for post-analysis.

4. Experiments

To evaluate the performance of the DZ50 platform, we have implemented both applications in TinyOS 2.x and we have conducted experimental measurements of the energy consumption using an oscilloscope. The same measurements have been also performed on two other sensor platforms: MicaZ (for the SPS application) and TelosB (for the PIS application).

The hardware setup for the SPS application is shown in Fig. 3. We have employed the MAG3110 magnetometer[11] to measure the magnetic field around a parking spot which gives an indication about its occupancy. This chip gives measurements on the three axes and needs the establishment of an I2C bus. Note that we needed to use the MDA300 extension board to get access to the I2C pins of MicaZ and connect the MAG3110 chip. Fig. 4 shows the hardware

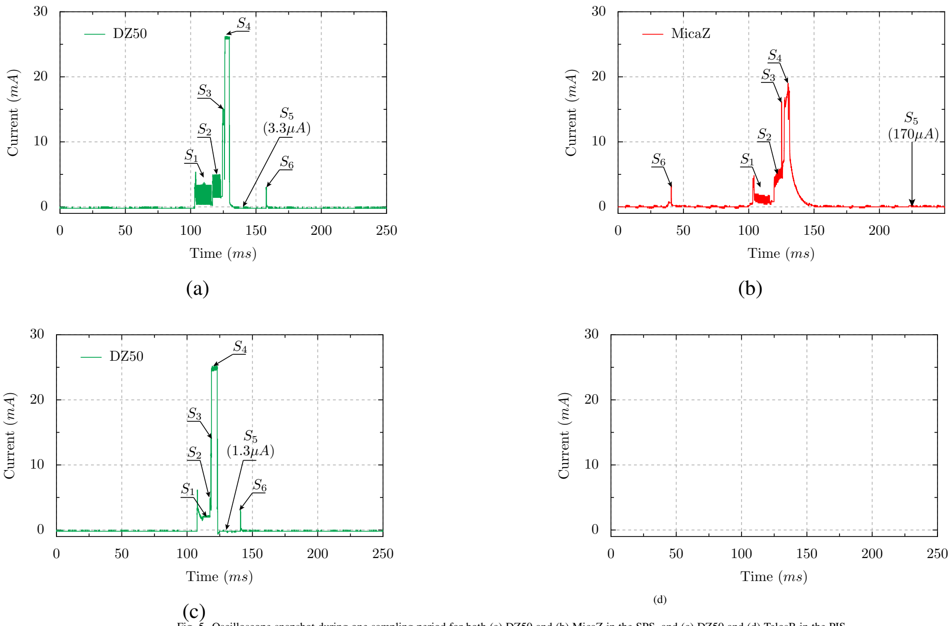


Fig. 5. Oscilloscope snapshot during one sampling period for both (a) DZ50 and (b) MicaZ in the SPS, and (c) DZ50 and (d) TelosB in the PIS. The different states are: S_1: sensor activation, S_2: data reading, S_3: radio starting, S_4: radio sending, S_5: sleep and S_6: timer interrupt.

setup for the PIS application. We have connected DZ50 and TelosB to the EC-5 soil moisture sensor[12] which determines the volumetric water content of the soil by measuring its dielectric constant. This equipment needs an excitation between $2.5V$ and $3.6V$ for a short duration (about $10ms$) and responds with an analog signal corresponding to the measured value. Since this device does not provide a sleep mode, we used a simple NPN transistor as a switch to turn it on when necessary.

Both applications share the same execution pattern, with some differences related to the actual need of the application. Basically, each mote periodically checks its sensing device and then sends the captured data to the base station for processing. Between two sampling periods, the mote is inactive and switches off its active hardware components, such as the sensors and the transceiver. Typically, this period (*sampling period*) would be at the order of no less than few seconds in such applications.

The execution snapshot of one sampling period (for both applications) is illustrated in Fig. 5. Six energy levels can be distinguished, which correspond to the different states of the application $\{S_1, \ldots, S_6\}$. Only the two first states differs between the two applications. The description of the execution is as follows. First, the mote is woken-up from its inactivity period and requests its sensing device to obtain a measurement (S_1). In the case of the SPS application, the application interacts with the MAG3110 magnetometer to sense the magnetic field. The sensor performs 16 samples, computes their average before returning to standby mode. However, in the PIS application, the EC-5 sensor is just turned-on with a $3.3V$ excitation during a period of $10ms$.

After the sensing period, the program retrieves the measured data (S_2). For SPS, the result is in a digital format and six bytes (2 bytes per axis) are retrieved using an I2C serial communication. For PIS, the data offered by the EC-5 sensor is analogical and thus the micro-controller uses its ADC to read it. After the second step, the application turns on the transceiver (S_3) and sends the data to the base station (S_4). Finally, the mote stops its transceiver and enters in sleep state (S_5). Since the asynchronous timer in both ATmega328P and ATmega128L are 8 bits registers, an overflow occurs every $225ms$, which explains the short interrupt activities indicated by the state S_6. However, the MSP430 micro-controller used in TelosB has a 16 bits timer register that allows it to overflow every $2s$.

We have measured the current consumption, c_i, and the duration, δ_i, of each state, S_i. The obtained results are summarized in Table 2. To estimate the application lifetime, we first define an approximation of the consumed energy

Table 2. Energy consumption levels of DZ50, MicaZ, and TelosB platforms.

	S_1 Sensing	S_2 Reading	S_3 Radio on	S_4 Radio tx	S_5 Sleep	S_6 Timer intrpt.
DZ50 SPS						
Duration δ_i	$13.5ms$	$7ms$	$0.312ms$	$4.2ms$	$225ms$	$0.14ms$
Avg current c_i	$2.2mA$	$3.25mA$	$14mA$	$26.5mA$	$3.3\mu A$	$3mA$
MicaZ SPS						
Duration δ_i	$13.5ms$	$6ms$	$2.5ms$	$2.5ms$	$225ms$	$0.27ms$
Avg current c_i	$1.2mA$	$4.6mA$	$10.2mA$	$18.5mA$	$170\mu A$	$4mA$
DZ50 PIS						
Duration δ_i	$9.762ms$	$0.696ms$	$0.312ms$	$4.62ms$	$225ms$	$0.14ms$
Avg current c_i	$2.25mA$	$4.35mA$	$9.86mA$	$24.23mA$	$1.3\mu A$	$3mA$
TelosB PIS						
Duration δ_i	$9.74ms$	$17.38ms$	$2.65ms$	$2ms$	$2s$	$0.14ms$
Avg current c_i	$1.39mA$	$1.72mA$	$3.98mA$	$19.12mA$	$13\mu A$	$2mA$

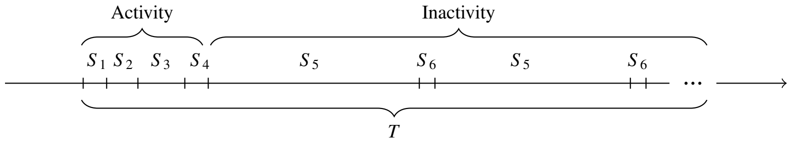


Fig. 6. Timeline of the different application steps.

$\mathcal{E}(T)$ during one sampling period of duration T as follows:

$$\mathcal{E}(T) = \sum_{i=1}^{4} c_i\delta_i + \frac{T - \sum_{i=1}^{4} \delta_i}{\delta_5 + \delta_6}(c_5\delta_5 + c_6\delta_6) \tag{1}$$

The intuition behind this approximation is depicted in Fig. 4 which describes the time division of a sampling period T. We can distinguish between two parts. The first part, starting from state S_1 to S_4, represents the activity period during which the mote retrieves the measurements and send them to the base station. During this period, the total amount of electric charge consumed is $\sum_{i=1}^{4} c_i\delta_i$, which represents the first part of $\mathcal{E}(T)$. During the inactivity period, the mote enters in sleep mode and is activated periodically by the timer interrupt until reaching the end of the sampling period T. An approximation of the number of activations is given by $\dfrac{T - \sum_{i=1}^{4} \delta_i}{\delta_5 + \delta_6}$, and for each activation period the mote consumes $(c_5\delta_5 + c_6\delta_6)$, which explains the second part of $\mathcal{E}(T)$.

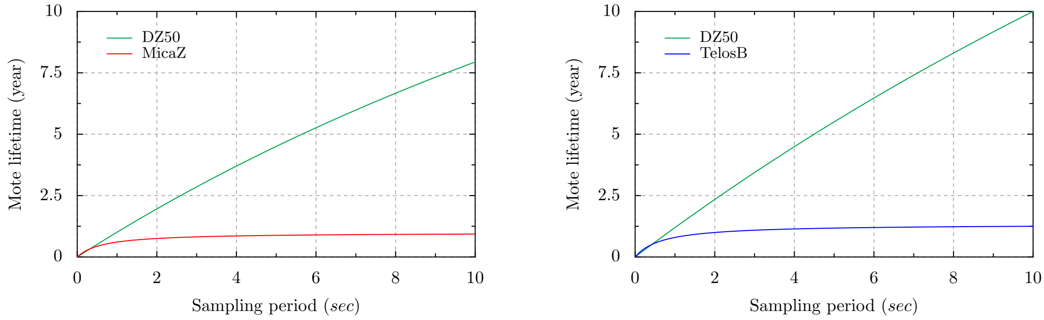
Therefore, given a battery charge, μ, the mote lifetime, say $\Phi(T)$, for an application with period, T, can be written as follows:

$$\begin{aligned}\Phi(T) &= \frac{\mu T}{\mathcal{E}(T)} \\ &= \frac{\mu T}{aT + b}\end{aligned}$$

where the linear model parameters (a, b) are empirically estimated using (1) and Table 2. The obtained values are as follows, depending on the target application:

- SPS:
  - $(a, b) = (5.16 \times 10^{-6}, 1.67 \times 10^{-4})$ for DZ50,
  - $(a, b) = (1.75 \times 10^{-4}, 1.11 \times 10^{-4})$ for MicaZ.
- PIS:
  - $(a, b) = (3.16 \times 10^{-6}, 1.42 \times 10^{-4})$ for DZ50,
  - $(a, b) = (1.3 \times 10^{-4}, 8.81 \times 10^{-5})$ for TelosB.

Based on this formula, the different system lifetimes are plotted in Fig. 7 as a function of the sampling period, when they are powered with battery of $1500mAH$. The results demonstrate the superiority of DZ50 platform over



(a) (b)

Fig. 7. Variation of the mote lifetime vs. the sampling period for (a) the SPS and (b) the PIS.

MicaZ and TelosB, starting from as a low sampling period as $1s$. DZ50 can extend the mote lifetime to more than 7 years for a $10s$ sampling period, while that of MicaZ and TelosB does not exceed 1 year.

5. Conclusion

In this paper we have presented DZ50, a wireless mote characterized by its very low energy consumption during inactivity periods. This important feature allows DZ50 to be a good platform for low sampling monitoring applications such as *Smart Parking* and *Precision Irrigation Systems*. The proposed platform have been empirically evaluated and compared with two platforms largely used by the research community, TelosB, and MicaZ. The comparison evolves typical settings of the two representative low sampling applications. Experimental results demonstrate its efficiency compared to both motes in terms of augmenting the battery lifetime where an a gain of more than 700% has been obtained making the system available for more than 7 years.

References

1. Akyildiz, I.F., Su, W., Sankarasubramaniam, Y., Cayirci, E.. Wireless sensor networks: A survey. *Comput Netw* 2002;**38**(4):393–422.
2. Ingelrest, F., Barrenetxea, G., Schaefer, G., Vetterli, M., Couach, O., Parlange, M.. Sensorscope: Application-specific sensor network for environmental monitoring. *ACM Trans Sen Netw* 2010;**6**(2):17:1–17:32.
3. Kang, J., Lee, S.C., Kim, H., Kim, S., Culler, D., Jung, P.M., et al. High-fidelity environmental monitoring using wireless sensor networks. In: *Proceedings of the 11th ACM Conference on Embedded Networked Sensor Systems*; SenSys '13. 2013, p. 67:1–67:2.
4. Memsic, . Micaz sensor mote specifications. `http://www.memsic.com/wireless-sensor-networks/MPR2400CB`; Accessed 2014.
5. Memsic, . Telosb sensor mote specifications. `http://www.memsic.com/wireless-sensor-networks/TPR2420`; Accessed: 2014.
6. Propst, J.A., Poole, K.M., Hallstrom, J.O.. An embedded sensing approach to monitoring parking lot occupancy. In: *Proc. of the 50th Annual Southeast Regional Conference*; ACM-SE '12. 2012, p. 309–314.
7. Mathur, S., Jin, T., Kasturirangan, N., Chandrasekaran, J., Xue, W., Gruteser, M., et al. Parknet: Drive-by sensing of road-side parking statistics. In: *Proc. of the 8th International Conference on Mobile Systems, Applications, and Services*; MobiSys '10. 2010, p. 123–136.
8. Wang, Y., Huang, L., Wu, J., Xu, H.. Wireless sensor networks for intensive irrigated agriculture. In: *Proc. of the 4th IEEE Consumer Communications and Networking Conference*; CCNC '07. 2007, p. 197–201.
9. Pierce, F.J., Elliott, T.V.. Regional and on-farm wireless sensor networks for agricultural systems in eastern washington. *Comput Electron Agric* 2008;**61**(1):32–43.
10. FAO, . Water management : Towards 2030. `http://www.fao.org/ag/magazine/0303sp1.htm`; 2003.
11. Freescale, . Mag3110 triple axis magnetic sensor. `http://www.freescale.com/webapp/sps/site/prod_summary.jsp?code=MAG3110`; Accessed: 2014.
12. Decagon, . Ec-5 soil moisture. `http://www.decagon.com/products/soils/volumetric-water-content-sensors/ec-5-soil-moisture-small-area-of-influence`; Accessed: 2014.