INSIGHT: Internet-Sensor Integration for Habitat Monitoring

Murat Demirbas Ken Yian Chow Chieh Shyan Wan
Dept. of Computer Science & Engineering
University at Buffalo
Buffalo, NY 14226
{demirbas, kenchow, chiehwan}@cse.buffalo.edu

Abstract

We present our experience with designing, developing, and deploying of an Internet accessible wireless sensor network for monitoring temperature, humidity, and illumination of a controlled environment. The design goals for our system, INSIGHT: INternet-Sensor InteGration for HabitaT monitoring, are extended deployment lifetime, remote querying and configuration, ease of deployment, and reliability. We present results from our deployment of INSIGHT in a greenhouse.

1. Introduction

Wireless sensor networks (WSN) are useful in many monitoring applications [7, 9, 10]. As it is becoming increasingly more feasible to deploy wireless sensor nodes for ubiquitous and high-fidelity monitoring, WSN has the potential to improve Supervisory Control and Data Acquisition (SCADA) [18] systems used for monitoring and control of a plant in industries such as telecommunications, water and waste control, energy, and transportation.

As a step towards the development of an industrial strength monitoring system, we investigate monitoring of a controlled, small environment via WSN. We see the following as the core set of requirements.

1. Energy efficiency: The sensor nodes should not need refreshing of batteries for at least 6 months.
2. Remote querying and reconfiguration: It should be possible to query data and to reconfigure the monitoring parameters via the Internet.
3. Ease of deployment: The client should be able to deploy the system without any special configuration.
4. Reliability: The data should be real-time and high-fidelity. The system should be available 99% of the time and should recover from a crash quickly.

Our Contributions. Here we discuss our experience with *designing, developing, and deploying of a system*, INSIGHT: *INternet-Sensor InteGration for HabitaT monitoring, that fulfils the above requirements.*

To enable energy efficiency and ease of deployment we use a single-hop network for INSIGHT. Without the need for forwarding messages from other nodes, the nodes do not need to stay awake or coordinate to wake up. In our case, each node occasionally wakes up only to send a report to the master (the base station) and spends the rest of the time in sleep mode to save energy. We also implemented a “delta reporting” technique [16] that enables both fast reaction time to changes in sensor readings as well as energy-efficiency when sensed values do not differ much from previous readings. INSIGHT can achieve about 6 months lifetime (using standard AA batteries) by sampling sensors every minute. Also due to our single-hop network decision, the deployment of the network is as easy as turning on a node and dropping it some place for monitoring.

To enable remote querying, we maintain a webserver and an SQL database at the base station. Users simply type in the base station’s web address into their internet browser to query for data. Data is available for extraction through an XML front-end on the webserver or in the form of a TinyML query [2]. Using the website, sensor data from motes can be visualized, plotted as a graph and compared with each other. Moreover, users can subscribe to get email alerts when a sensor reading has exceeded a threshold configured via the web interface. The client can also reconfigure the monitoring parameters, such as the sampling frequency and the delta thresholds, through the web interface. INSIGHT was deployed in a greenhouse and the website is accessible at http://INSIGHT.podzone.net for querying.

In order to maintain a high level of reliability, we keep the system as simple as possible. We use a watchdog timer for the nodes and the laptop to recover from crashes and remote login capability to change webserver settings.

We present results from the deployment of INSIGHT in the Dorsheimer Greenhouse at the University at Buffalo. Through the deployment, we demonstrate INSIGHT's energy efficiency, reliability, ease of use, and remote querying capabilities. Our deployment at the greenhouse surfaced some concerns with overheating that may be harmful to the experiments the scientist has been running on the tobacco plants.

Related work. Due to the industrial significance of greenhouse monitoring, Crossbow offers a WSN solution to this end that aggregates sensed data in a database and uses MATLAB as the front-end for querying [17]. In contrast to Crossbow's solution, INSIGHT is fully Internet integrated and enables querying and graph generation via a web interface, as well as allowing reconfiguration of the monitoring parameters, and subscriptions to alerts based on specified thresholds.

The idea of using a tiered architecture is not new. TENET [19] suggests an architecture where sensor nodes are responsible only for data gathering, and more powerful tier-2 nodes, such as Stargates [17], are responsible for storing and managing the data as well as retasking the WSN. In contrast to TENET which allows multihop WSN, we emphasize the benefits of using a single-hop WSN architecture as it simplifies the WSN software significantly as well as prolonging the lifetime of nodes.

TinyDB [16] introduced the delta reporting technique we use for energy conservation. TinyDB uses a multihop WSN and coordinates the sleep patterns of the nodes for data exfiltration from the WSN. However, TinyDB does not run on mica2 or TelosB mote architectures. In contrast to TinyDB, INSIGHT provides Internet integration at the basestation.

Outline of the rest of the paper. After presenting the system architecture in Section 2 we discuss our methods for energy efficiency in Section 3, methods for reliability in Section 4, and Internet integration in Section 5. In Section 6, we present results from our deployment in the greenhouse, and we conclude the paper in Section 7.

2. System Architecture

INSIGHT follows a simple system layout (refer to Figure 1 for a system overview). All sensor nodes send their sensor readings to the base station, and the base station serves those data to users via the Internet.

Our sensor network provides four sensor readings: humidity, temperature, photosynthetically active radiation (PAR) light, and internal voltage. The data sample from the PAR sensor is the light intensity that is useful for plants and the internal voltage reading is used for monitoring the battery life on the mote itself.

The base station is responsible for aggregating all the sensors data, converting the raw data into meaningful values and storing them into a database. Users can access those data by visiting a webpage on the server or running an XML query and then extracting the result from the base station.

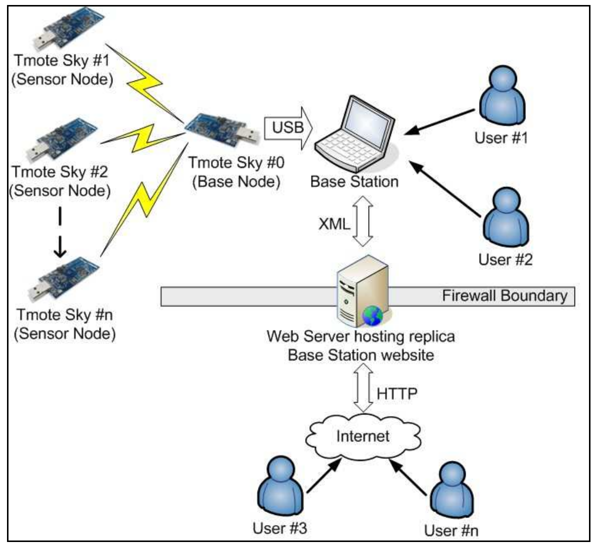

Figure 1. INSIGHT system overview

2.1. Sensor network

Our sensor network uses the Tmote-Sky hardware manufactured by MoteIV [6]. The sensor nodes (motes) come in matchbox-size printed circuit boards. Each mote has an 8MHz Texas Instruments MSP430 microcontroller (10k RAM, 48k Flash) and a 250kbps 2.4GHz, IEEE 802.15.4 Chipcon CC2420 Wireless Transceiver which is Zigbee ready [13]. The field-deployed motes also have integrated humidity, temperature, and light sensors on board. Humidity and temperature is measured using Sensirion's SHT11 [14] sensor while light is measured using Hamamatsu's S1087 PAR light sensor [15]. Motes use two standard AA batteries for power.

To keep the architecture simple, INSIGHT uses single-hop communications between motes and the base station. The CC2420 radio has communication range of approximately 100 meters [6], which is well beyond the size of most of our target deployment environment. We use the TinyOS version 1.1.14 distribution [1] as a development framework for the motes firmware. To have the motes sample and send the data via wireless communication, we modified the Oscilloscope program provided by Moteiv to optimize the sensor reading and sampling rate to fit our application. TinyOS's communications module for Tmote uses the Berkeley Media Access Control (B-MAC) [4].

2.2. Base station

The base station provides the interface between users and the WSN by collecting sensor readings, managing the database, and enabling remote querying and reconfiguration as well as supporting subscriptions to alerts. We chose to use a portable computer as it is simpler to develop and run the base station using a portable computer as opposed to a palm device or application-specific gateway [17]. One of the motes connects to the USB port of the laptop and act as a gateway. The laptop came equipped with an 802.11g wireless card and is able to connect to any nearby wireless access point that provides access to the Internet. The laptop is setup to use Dynamic Host Configuration Protocol to obtain an IP address automatically either from the wireless access point or through the Ethernet. Figure 2 lists the software running on the basestation.

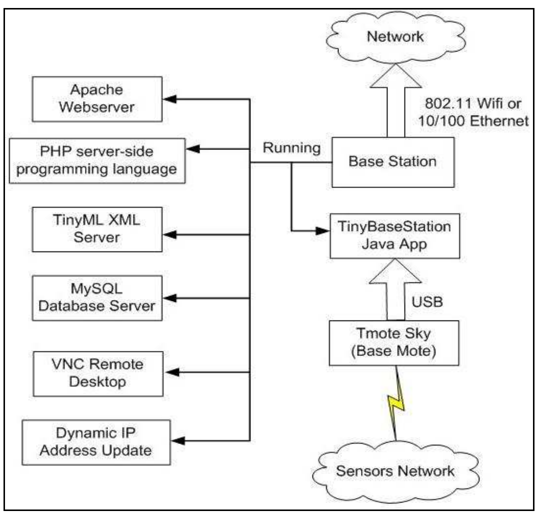

Figure 2. Basestation system overview

The base station connects to the greenhouse network which is part of a campus-wide network operated by UBIT. Due to security issues, all incoming connections from outside the UBIT network are blocked by default. There are only some designated webservers that can be accessed from outside the network. In order to serve a website to the public, the webpages must be placed on one of the designated webservers. To get around this obstacle, we continued to run a webserver on the base station which is only accessible by users within the UBIT network. We then have a replica of the website on one of the designated webservers accessible publicly. This website obtains data periodically from the base station via a simple script that uses XML queries. We set the update frequency to be hourly, so as not to generate too much traffic on the network. While posing as a problem initially, this issue turned out to showcase one of the main strengths of INSIGHT: ease of deployment. Setting up the replica turned out to be trivial, since nothing needed to be changed except the database connection string.

3. Energy Efficiency

One of the goals of INSIGHT is to prolong the lifetime of the sensor nodes by putting them to sleep when they are idle. Recent versions of TinyOS (version 1.1.14 and onwards) have support for a Hardware Presentation Layer (HPL) for the CC2420 radio. The HPL power management module puts the mote to sleep when certain conditions are met: 1) the radio is turned off, 2) SPI (Serial Peripheral Interface) interrupt is disabled, 3) task queue is empty, and 4) high speed clock's output-compare-interrupts are disabled [1]. Our system uses the GenericComm module, which provides the HPL power management control interface. When a mote is idle, we explicitly turn off its radio for the HPL to put the mote to sleep. The result with the power management is significant as seen in Table 1. When the radio is off, the current consumption is 0.27 milliamps with HPL versus 22mA without HPL, indicating a saving by a factor of 100. Also when the radio is on, consumption with HPL is 20mA compared to 29mA without HPL. Thanks to our single-hop network, when a mote does not have any data to report it can sleep without the need to wake up to forward messages from other nodes.

Table 1. Comparison of Energy Consumption with different firmware

	Basic code	With HPL
Radio On	29 mA	20 mA
Radio Off	22.6 mA	0.27 mA

In addition to the HPL power saving module, INSIGHT has also implemented the delta reporting mechanism (similar to TinyDB's technique [16]) to prolong the sensor's active lifetime. In each duty cycle, the sensor calculates the changes between the current sample and the cumulative average; if the change is within the acceptable delta range, the mote goes back to sleep without reporting to the base station. Otherwise, the mote will send the sampled reading to the base station. Each duty cycle is set to 1 minute, and after 20 rounds of duty, the cumulative average readings are reported to the base station as a heartbeat message and the average is reset. The improvements due to the delta reporting are presented in Section 6.

By comparing a sensor reading to its average, INSIGHT is highly sensitive to a change in sampled readings. By default, INSIGHT is set to report sample readings if the humidity reading differs by more than 1% to 2% from the average, temperature reading differs by roughly 0.2 Celcius, PAR data changed by more than 2 Lux or a 0.03 volts change in internal voltage reading.

4. Reliability

Here we discuss our experience with the reliability of INSIGHT in our deployment.

During one of our early experimental deployments we noticed that one of the sensor nodes stopped sending messages to the base station. More specifically, we found that even though the node wakes up and samples data, it fails to transmit any message. We believe this is caused by a "Transmission Pending" bit not being reset after transmission is done. We confirmed this by observing that an LED was on all the time, when it should be off after a transmission is done. We addressed this issue by using a timer to reset the "Transmission Pending" bit, in case it fails to be cleared by the "Send_Done" event. We have not witnessed the same type of failure after this fix.

We also implemented a watchdog timer to recover from cases where a mote 'freezes' and fails to perform any further tasks. This timer is set to run when the mote is sleeping; it has to be reset every time it wakes up and queries the onboard sensors. If for any reason this timer is left running over a specific period of time, its overflow interrupt will force a soft reset on the motes.

In our deployment, we have not witnessed the webserver, database server or the operating system crash. However, we have provided a watchdog timer script that resets the TinyBaseStation application, the webserver and the database in case they become unresponsive. We do not expect any catastrophic risk in doing this, since we will lose at most 5 sensor data consecutively (it takes 3-5 minutes to reset the servers).

5. Internet Integration

One of our major contributions with INSIGHT is the integration of a WSN with the Internet. Our focus here is on the ease of use and ease of deployment.

Ease of deployment: INSIGHT is designed to operate and provide environmental data to the outside world with only minimal user intervention. The base station is setup such that all essential applications launch themselves automatically on startup. The Serial Forwarder and Base Station software runs upon boot up, the Apache webserver begins serving webpages immediately and users can locate the webpage of the base station by navigating to a dynamic DNS address [8] which detects and updates that address to point to the current IP address obtained by the base station's network interface. MySQL server is installed on the base station to provide a database server for storing motes information (e.g. description, location, etc.) and sensors data. Sensor data is timestamped as it arrives in the database. The system can basically be up and running by just turning on all the motes and the base station.

Ease of use: There are two ways for a user (a human or a program) to access and query the WSN. The first is by visiting the website on the base station to retrieve data. Alternatively, we also provide an XML interface for data extraction. For the XML part, data can be queried using TinyML [2] or through the web interface. The results of the query are placed on an XML file available via the webserver.

In line with keeping the user interface simple but powerful, the website+ is divided into three parts:

1. Graphical Overview – Provides access to the data by using graphs++.
2. Tactical Overview – Provides real-time access to the data in a top-view image.
3. Query Wizard – The wizard asks a question and the user select the options desired. The data is then returned based on the user's choice.

By using any of the 3 views on the website, we believe INSIGHT can cater to a wide range of requests from a simple overview to detailed querying of each sensor.

6. Deployment result

We have deployed INSIGHT in the Dorsheimer Greenhouse at the University at Buffalo to help a scientist closely monitor the environmental conditions in the greenhouse while she is running experiments on the growth rate of tobacco plants. Mote deployment locations were picked by the laboratory personnel. Due to the relatively small size of the bay, there were no disconnected or unreliable links between the motes and the base station.

After a twenty-day deployment, the battery life on the motes was still very good. Average transmission frequency is about 23 packets per hour, varying from 12 packets to 58 packets per hour at various times of the day. Figure 3 shows the average transmission of a mote per hour. The calculation is done by taking the average of all motes across a period of 7 days.

We observed a drop of about 0.13 volts over the period of 20 days. Since motes are able to survive until the operating voltage falls below 1.8V [6], we can afford

+ The website design is due Luka Cvrk [http://www.solucija.com]
++ Graphs generated using JPGraph [http://www.aditus.nu/jpgraph]

a drop of 1.2V from a fresh set of batteries. 1.2V divide by 0.13V multiply with 20 days will yield 184 days. This is in line with our expectation that a mote can survive for 180-200 days before running out of batteries+++. A comparison of the motes with and without Delta Monitoring is shown in Figure 4. We observe that Delta Monitoring improves the lifetime significantly.

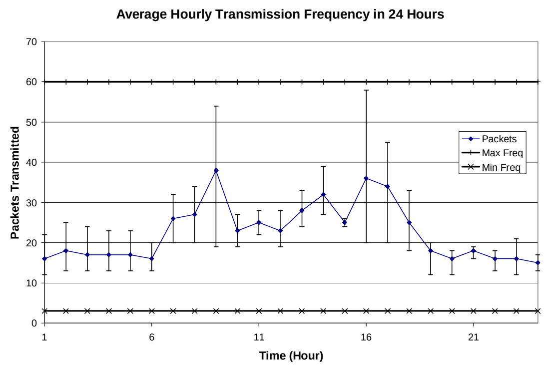

Figure 3. Transmission frequency per hour

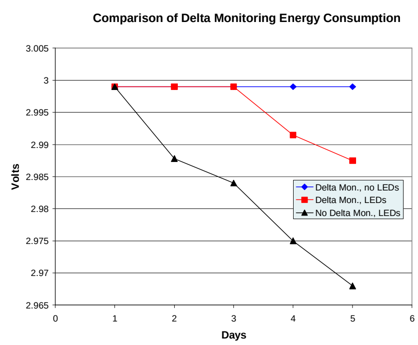

Figure 4. Voltage comparison

Table 2. Average, Low & High Temperature in the Greenhouse from Jan 10 through Jan 15

	Average (ºC)	Low (ºC)	High (ºC)
1/10/2006	26.39	18.46	37.29
1/11/2006	30.09	18.20	46.67
1/12/2006	34.86	19.88	49.56
1/13/2006	28.02	20.47	37.86
1/14/2006	23.53	15.48	31.96
1/15/2006	29.62	10.61	48.34

+++ In the worst case scenario (i.e. a volatile environment), the motes transmit data every minute: For approximately one second every minute, each mote turns its radio on while the radio is off for the rest of the time. This amounts to a 0.6mA average current consumption per minute and will give roughly 4800 hours service time (about 200 days) on two AA-sized batteries with 2900 mAh capacity each. Furthermore, a single Lithium battery pack with 8500 mAh capacity can provide up to 14000 hours service time (more than 18 months) with the same current consumption [12].

By placing the motes strategically, the scientist managed to identify locations that are wasting heat. Comparing a mote placed in the center and a mote placed about 5 feet higher (Figure 5), there was already a 2 degrees Celsius difference. Besides the large difference within the greenhouse, overall temperature was also fluctuating significantly. The ideal temperature should be around 27 ºC, but we observed variations from as low as 15 ºC to as high as 49 ºC (Table 2). According to the scientist, the plants within the greenhouse may be at risk if left above 40 ºC for over 1 hour. The reason for the overheating is due to the air conditioning being switched off by facility management during winter months.

Also of interest is the observation that temperature and humidity in the greenhouse are often inverse of each other (refer to Figure 6). Given a temperature, the humidity is highly predictable and vice versa. At certain times though, temperature and humidity appear to both drop. This is due to the heater being turned on, which reduces humidity dramatically.

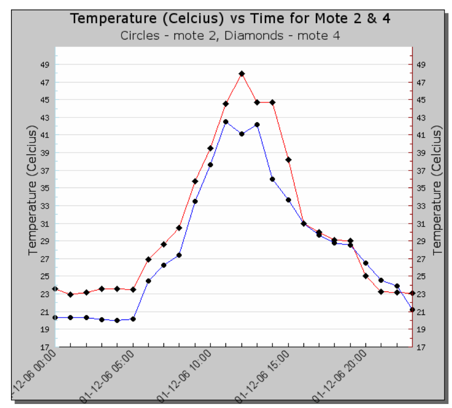

Figure 5. Mote 2 (center) vs Mote 4 (elevated) temperature comparison on Jan 12

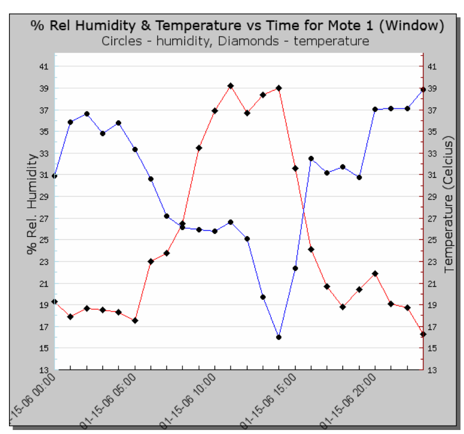

Figure 6. Humidity & Temperature comparison on Jan 15 for Mote 1 (Window)

The scientist using INSIGHT to monitor the greenhouse found that it was easy to deploy and use. She was especially happy with INSIGHT's thorough and high-fidelity monitoring data which she can study and use to make her case to the facility management about problems she has been facing with the operation of the greenhouse.

7. Concluding Remarks

In this paper we reported on our experience with designing, developing, and deploying of INSIGHT. INSIGHT enables remote querying and reconfiguration of a WSN via Internet and subscribing to alerts based on user specified threshold over the web. INSIGHT is energy-efficient; the batteries on the sensor nodes do not need replacing for about 6 months (and with Lithium batteries, over a year). Furthermore, the sensor network's fidelity (i.e. sensitivity to environmental changes) can be set so that it is responsive yet energy efficient. Finally, INSIGHT does not require pre-configuration, turning on the sensor nodes and the base station is enough for deployment. For environments where a PC is available, the PC can be used as a base station, reducing the cost of INSIGHT to just the cost of sensor nodes, which is currently about $100 each. With all these to its advantage, we hope that INSIGHT can serve as a stepping-stone for industrial quality WSN monitoring system that can help improve on existing SCADA systems.

We believe that a single-hop network architecture is a better choice than providing multi-hop capability in the WSN. Using single-hop network architecture improves energy-efficiency as nodes do not need to stay awake to receive broadcasts from other nodes for forwarding, facilitates the development as implementation of complex services (such as routing, time-synchronization, and coordination) can be avoided, and enables reliability through simplicity. Therefore, for monitoring a large area where one single-hop WSN is inadequate, we believe that, from the holistic system design perspective, using multiple base stations with single-hop WSN associated ith each is a better choice than using a single base station with a multi-hop WSN.

In our experience, enabling Internet accessibility enhanced the usability of the WSN greatly. As most users are familiar with the Internet, they find our web-based user interface easy to understand. Also due to the ubiquity of Internet, users can conveniently access INSIGHT functionality such as querying, reconfiguration, and subscribing to alerts from anywhere, anytime.

In future work, we will improve INSIGHT's functionality by integrating intrusion detection and actuator/control mechanisms using off-the-shelf hardware components. To improve energy-efficiency further, we will investigate a predictive monitoring technique similar to that in [3]. Finally, we will consider hardware-based watchdog timer solutions to address the case of an OS crash at the base station. Since all sensor data is saved upon arrival at the base station, and since the INSIGHT software starts automatically on boot-up a single-reset of the base station could be sufficient for our purposes.

Acknowledgment: We would like to thank Dr. Margaret Hollingsworth for collaborating with us to use her greenhouse bay at the Dorsheimer Greenhouse Laboratory as a test site, and for insightful comments that helped us improve our system.

8. References

[1] TinyOS Tutorial: www.tinyos.net
[2] N. Ota, WTC Kramer, "TinyML: Meta-data for Wireless Networks", Technical Report, University of California, Berkeley, Computer Science Dept, 2003.
[3] S. Goel and T. Imielinski, "Prediction-based monitoring in sensor networks: Taking lessons from MPEG", Computer Comm. Review, 31(5), Oct. 2001.
[4] J. Polatre, J. Hill, and D. Culler, "Versatile low power media access for wireless sensor networks", In ACM SenSys, 2004.
[5] Ilyas, editor, *Handbook of Sensor Networks:. Compact Wireless and wired sensing systems*, CRC Press, 2003.
[6] Moteiv Tmote Brochure/Datasheet: www.moteiv.com
[7] B. Tolle, J. Polastre, R. Szewczyk, D. Culler, N. Turner, K. Tu, S. Burgess, T. Dawson, P. Buonadonna, D. Gay and W. Hong, "A Macroscope In The Redwoods", In ACM SenSys, 2005.
[8] PL Tsai, CL Lei and WY Wang, "A remote control scheme for ubiquitous personal computing", In IEEE International Conference on Networking, Sensing and Control, 2004.
[9] A. Mainwaring, J. Polastre, R. Szewczyk, D. Culler, and J. Anderson, "Wireless Sensor Networks for Habitat Monitoring", In WSNA, Georgia, September 2002.
[10] J. Burrell and T. Brooke and R. Beckwith, "Vineyard Computing: Sensor Networks in Agricultural Production", In IEEE Pervasive Computing, 2004.
[11] Energizer e2 datasheet: data.energizer.com
[12] Tadiran Lithium Ion datasheet: www.tadiranbat.com
[13] Chipcon CC2420 datasheet: www.chipcon.com
[14] Sensirion SHT11 datasheet: www.sensirion.com
[15] Hamamatsu S1087 datasheet: www.hamamatsu.com
[16] S. Madden, J. Hellerstein and W. Hong, "TinyDB: In-Network Query Processing in TinyOS", Intel Research, IRB-TR-02-014, , 2002.
[17] Stargate datasheet: www.xbow.com
[18] L. Tan and K. Taylor, "Mobile SCADA with Thin Clients - A Web Demonstration", ICITA, Bathurst, Australia, 2002.
[19] R. Govindan, E. Kohler, D. Estrin, F. Bian, K. Chintalapudi, O. Gnawali, S. Rangwala, R. Gummadi and T. Stathopoulos, "Tenet: An Architecture for Tiered Embedded Networks", CENS Technical Report 56, 2005.