SeeMote: In-Situ Visualization and Logging Device for Wireless Sensor Networks

Leo Selavo, Gang Zhou, John A. Stankovic
Department of Computer Science
University of Virginia, Charlottesville, VA 22904
{selavo, gz5d, stankovic}@cs.virginia.edu

ABSTRACT

In this paper we address three challenges that are present when building and analyzing wireless sensor networks (WSN) as part of ubiquitous computing environment: the need for an in-situ user interface, a data logger, and a power consumption meter. Solutions for the above have been presented using laptops, personal digital assistants (PDA), onboard flash memory chips of limited size (usually 1MB), and laboratory test equipment. All of them have a good utility for the right applications. However, considering a certain variety of WSNs, where size, battery life, and cost are crucial, none of the above solutions is satisfactory. In this paper we present a compact, lightweight, low power, and low cost multimodal sensor module *SeeMote* that meets the stated challenges, and is compatible with the popular MICAz mote. Our module has the following components: (1) a graphical user interface component that combines a color liquid crystal display (LCD) and 5-way buttons, (2) a power meter component that is reconfigurable for attaching various low-power devices, and (3) a data logger component that is interfaced to a removable secure digital (SD) or multimedia memory card (MMC). The module dimensions are 34x58x12mm. This paper describes the hardware and software design and experiences while developing and using the device. The device is evaluated by comparing its parameters and functionality to laptop and PDA solutions. We conclude that *SeeMote* is preferred for certain WSNs, such as very large scale, difficult to reach, and wearable WSNs. We also present several applications that use the LCD module, such as the portable frequency spectrum analyzer and remote sensory data display device.

1. INTRODUCTION

In the ubiquitous computing world, where computing devices are seamlessly integrated with the environment, a wireless sensor network is a natural tool for monitoring the environment and reporting the essential data and events as per the application. The primary functions of the wireless sensor network (WSN) are the collection, processing, and delivery of sensory data. Usually the data is delivered to the main processing station or a database. However, there are scenarios when the user needs to see the data, diagnose or debug the wireless sensor network while being in the field. These scenarios demand a mobile device with immediate visual feedback to the user.

Often a sensor network can be observed in the field via a laptop computer or a personal digital assistant (PDA) that is connected to the network through a gateway device. However, laptops and PDAs have a limited power supply when mobile. This is due to the power hungry features such as high performance processing and displays. They are also rather big and costly for many types of WSN in which using a laptop or a PDA would be too inconvenient or expensive. Let us consider a few such networks:

- **Large Scale WSN**: Consider a large number of sensor nodes spread over a large area, perhaps, a mountain. One such WSN was deployed on a volcano and required several hours of hiking to reach and move across the deployment site [1]. This requires lightweight tools and long battery life.
- **Difficult to reach WSN**: Consider a network deployed in an area that is difficult to reach or traverse, such as forest or jungle. For example, there is a WSN deployed on a redwood tree [2]. Bringing a laptop for diagnostic purposes to the top of the tree would impair the navigation capabilities of the person. The size and the weight of the equipment is an issue.
- **Industrial WSN**: Consider a large industrial environment with many sensor nodes deployed over a large area. Some of the nodes are display devices that function as diagnostic hubs by presenting information about the sub-cluster of the network. They log data locally thus minimizing the communication overhead and power expenditure by the network as a whole. The administrator checks the network on site, and, perhaps, collects the removable storage cards with the logged measurements. By having several such nodes deployed requires them to be low cost.
- **Wearable WSN**: Consider a wearable body network that monitors vital signs of a patient [3] or a healthy person hiking at high altitude while monitoring the oxygen level in blood, heart rate, and other vital signs. Such applications need a user interface module that presents the current readings or generates an alert in case of emergency. But such a module can not be heavy and bulky, because it has to be unobtrusive to the wearer and to the surrounding environment. Long operation time is also an important requirement.

These systems need a user interface device that is lightweight, compact, has long operation time, and is low cost. Similarly, large amounts of data may need to be logged in the field because transmitting them over the network may take too much time and bandwidth, cost too much power, or is simply impossible because the wearable network is temporarily isolated from outside connections. In addition, if such a network needs to be debugged or evaluated from the perspective of power consumption, it might be too cumbersome to bring the conventional electronics test tools to the field because of their size, cost, and the question of powering them while they are in the field. However, the importance of performing the measurements after the network is deployed should not be disregarded because of the many factors that are hard to account for in the laboratory settings. For example, communication patterns over larger distances and through the environment with natural obstacles may require higher radio power or result in higher error rates and thus require retransmission. Temperature variations may change the power consumption of the electronics and voltage levels of the batteries. Therefore a compact, portable power measurement device is a valuable tool for measurements in the field.

We propose a solution called *SeeMote* that addresses all of the concerns mentioned above. *SeeMote* is a low-power, low cost sensor module for MICAz motes. It provides a graphical user interface, data logging capabilities, and a power consumption measurement feature.

Designs of motes that have a user interface capabilities have been presented before. For example, the Pluto mote as part of the CodeBlue project at Harvard provides input from a user and has limited feedback through a set of LEDs [4]. Other designs have a LCD screen that is built in the sensor node [5]. Our module differs from these designs in the following ways:

- **Compatible design**: the module is designed as an attachable sensor-board for the popular MICAz and Mica2 motes [6] made by Crossbow Technology Inc (Figure 1).
- **Multicolor graphical user interface**: the module has a multicolor LCD screen and five-way input buttons. It can use built-in video memory as a shared resource for the user interface or for temporary data storage by the application as necessary.
- **Removable storage**: The module provides interface for the popular secure digital (SD) and multi media (MMC) data storage cards. The removable storage media enables customizing the memory size according to the application and as an alternative way to transport the stored data between the mote and the data processing computer.
- **Power-meter functionality**: the module is capable of measuring electric power consumption by monitoring electric current and supply voltage for external low power devices. Thus, the module can diagnose power consumption on the LCD screen and log it in the removable storage.

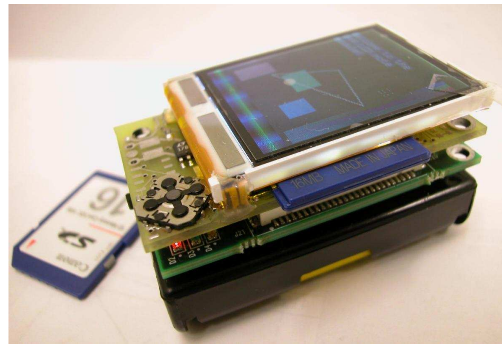


Figure 1: SeeMote - Multimodal Sensor Module

The rest of the paper is organized as follows: First, we discuss the hardware and the software components of the multimodal module. The subsequent section presents applications that are enabled by the SeeMote module followed by comparison of the SeeMote and the solutions involving laptops and PDAs. We conclude the paper with the discussion about the potential contributions of the module to wireless sensor networks.

2. HARDWARE

The SeeMote multifunction module was designed as a user interface node for sensor networks. It has a liquid crystal display (LCD) for information output and five way navigation buttons for user input. The module also has a removable memory storage interface that is compatible with the popular SD and MMC cards. Finally, it has a sensor module that is capable of power consumption measurements for external low-power devices, such as other WSN nodes (Figure 2). All these devices are interfaced to the Mica mote via the standard 51-pin bus connector. The data is transferred between the mote and the components using I2C, SPI, and custom data busses. We discuss the components in the following subsections.

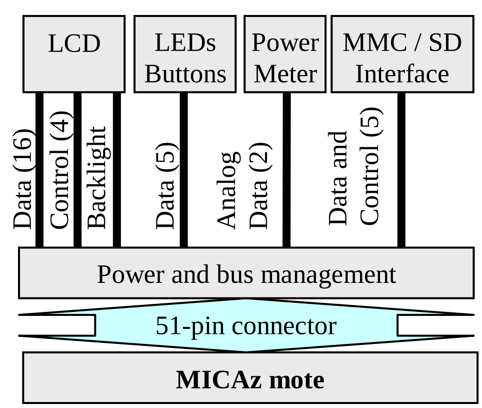


Figure 2: MICAz and Multimodal Sensor Module components

2.1. LCD Screen

The display component of the multimodal module has to be able to present information to the WSN user in a legible form at reduced size and reduced power consumption. We decided to use a color LCD display of size that would present a reasonable resolution and would fit to the size of MICAz mote. We found that the LTM018A600 LCD module fits the requirements. The LCD features 1.8 inches by the diagonal, has 128x160 pixels, and is 65K color capable [7]. The module has integrated 46KB video memory (RAM). The module is powered by 1.8-3V and is accessed by a 16 bit data bus and 4 control signals. Typical current at 3V according to the datasheet is 100-210µA for the digital power supply, 200-400µA for the LCD drive, and 15mA for the backlight when it is enabled. The LCD controller has two stand-by power modes that reduces power consumption, and therefore, constitutes a good choice for the low-power application.

The LED backlight integrated in the LCD requires a constant current power supply. Therefore, the LT1937 charge pump is added to our design. It delivers 15mA to the backlight at the typical conversion efficiency of 84% and can be enabled or disabled by the software application. When testing MICAz without the module, the current consumption was 30-35mA at 3V power supply. After attaching the LCD module with the backlight on the current increased to 110mA. The backlight charge pump is the major power consumer; therefore we attempted to lower the current consumption by decreasing the backlight brightness. This is done by changing the sense resistor used by the charge pump [8]. Eventually the resistor was changed from 6 to 27 Ohms resulting to 60mA current consumption by the whole system at reduced backlight brightness. The display was still readable in daylight, while the power consumption was cut almost in half.

The application has control over the digital and the liquid crystal drive power supplies of the module. Thus the application can preserve and access the video RAM while powering down the LCD drive and backlight circuits in order to minimize the power consumption. In this mode the memory data is accessible, i.e. the image can be pre-painted on the LCD or the video RAM can be used as temporary data storage.

The LCD module is controlled by the integrated Himax HX8302A and HX8609A chipset, which allows per-pixel access to the display. It has capabilities of defining two independent display windows and provides a list of logical graphical operations, such as *replace*, *and*, *or*, and *xor*, allowing for advanced graphical operations if desired [9]. The character generation for text output is not directly supported; therefore our software driver for the LCD module provides font generation and text output functions.

The communication to the LCD module is implemented using GPIO pins available on the 51-pin connector of the MICAz compatible motes. We use 8 PW and 8 ADC lines to deliver 16-bit data to the LCD. In addition we use four more GPIO lines for *chip-select*, *write*, *read* and *reset* signals of the LCD module. The LCD and backlight power are managed over I2C bus by ADG715 switch chip. The same chip also multiplexes two of the ADC bus lines between the LCD and the power-meter module. This choice was made due to the limited number of GPIO pins provided by MICAz interface and to implement the 16 bit LCD data bus efficiently.

2.2. User Input

The user interacts with the mote using a five way navigation interface, which is implemented as a five button module. The user can choose any of the four direction buttons and also press the select button, thus enabling efficient navigation through the user interface presented by the software. Two of the buttons are designed to generate hardware interrupts, which enables the user to bring the device out of the sleep state by pressing the button and thus to achieve a more efficient power management. The number of interrupt-capable buttons is limited by MICAz, which has only four interrupt lines available and reserves one of them for the radio chip interface, while another is reserved for the SD card interrupt. Adding a keyboard controller that allows interrupt sharing among all of the buttons would incur additional complexity, costs, and likely increase power consumption by the module.

In addition to the buttons the user interface module has three color LEDs that operate in two ways: they light up whenever the user presses a button or when activated by software providing feedback to the user as defined by the application. Thus, when simple communication is needed, the application communicates to the user by LEDs while the LCD is in sleep mode.

2.3. Removable Storage

Initially, we planned to use SD memory cards for removable storage because these are used for digital cameras, PDAs, MP3 players, and other portable electronics items and, therefore, very popular and widely available. Many computers have a SD card interface, which provides seamless WSN data sharing and transport through portable storage. However, it was found that using SD protocol requires a license which can increase the costs of development significantly [10]. However, looking further, we found that most SD cards are required to be compatible with the older MMC interface standard [11], which is open for public use to the best of our knowledge. Thus, we decided to provide hardware interface that is compatible with SD and MMC cards allowing both communications protocols, as decided by the application programmers. By providing the interface to these devices our module enables up to 4GB removable storage for the MICAz based WSN nodes.

The interface to the storage card is implemented via the SPI bus. The low level interface is very similar to the MICAz onboard flash memory chip interface, with which the bus is shared. The only difference is a dedicated chip-select pin used to address the SD card. In addition, the mote can turn the power on and off for the SD card and thus save energy or reset the card as appropriate.

2.4. Power-Meter

The task of the power meter module is to measure the energy consumed over time by an external device. This is accomplished by placing a small resistor in series with the power line of the device being tested and sampling the voltage drop, which is proportional to the current flowing through this resistor. The LT1787 high-side current sense amplifier is a good choice for sampling and amplifying the voltage drop due to its low-power characteristics and capability of handling the current in both directions, essentially allowing monitoring current in devices in which the current may change the direction, such as the systems with the rechargeable batteries. Voltage is measured in addition to the current to provide accurate calculation of the consumed energy.

The connector on the power meter board provides the power to the test device either from the same supply as the power meter or from an external source connected to the same connector. The latter option is preferred because then the power source is isolated from the measuring node and thus enables more accurate measurements.

The precision and rate of the measurements are limited by the ADC operation of the ATMega128 controller present in MICAz mote. The ATMega128 chip has 10-bit ADC and is capable of delivering up to 384 measurements per second (13 cycles of up to 200kHz per each measurement) [12]. This, however, does not include the software overhead, which depends on the processing and delivery of the measurements. For example, the power can be measured and logged to the removable storage, immediately viewed on the screen, or transmitted wirelessly, all of which add a delay before the next measurement.

3. SOFTWARE SUPPORT

In order to support the hardware we developed a software driver in NesC for TinyOS that takes care of initialization, operation, and power management of the LCD module. The driver is primarily based on components already included with TinyOS, such as I2C and SPI bus interface for the low level communication with the power management device. However, the LCD 16-bit data interface required implementation of a custom communications protocol that observes the timing of the control and data signals as per the integrated LCD controller specifications. The timing of the signals was obtained using TOSH_uwait() function present in the TinyOS library. After implementing the protocol we verified the timing with an oscilloscope and found out that some of the TOSH_uwait() calls can be removed without violating the LCD controller interface specifications and thus the data exchange speed increased.

3.1. Low Level Interface Support

The next task in the interface hierarchy was to create low level support procedures in the context of the TinyOS component interface, such as the LCD controller chip initialization command that sets the internal voltages, color gamma values, video RAM and graphics modes and awakens the LCD to the fully operational state. Other procedures of the driver interface are for entering or exiting sleep and standby modes as well as managing the backlight charge pump (Table 1). These procedures are used by the upper interface layer commands and simplify the LCD access routines for the developer considerably.

In addition we created several higher level components: (a) Mode switch that controls the power supply lines for the individual sub-components of the SeeMote module and switches the data bus between the LCD and the power-meter, (b) LCD driver that powers up and initializes the LCD module registers and provides a convenient interface to the LCD module, such as managing the LCD power, selecting color, and printing text and geometric primitives on the screen, and (c) an input driver that wakes up the mote and monitors the input button actions.

Low Level

- `command result_t reset();`
- `command result_t powerSetup();`
- `command result_t chipSetup();`
- `command result_t write(uint16_t idx, uint16_t data);`
- `command result_t writeGram( LcdColor_t* data, uint16_t len);`
- `command result_t writeGramAt(uint16_t idx, LcdColor_t* data, uint16_t len);`
- `command result_t read(uint16_t idx, uint16_t data);`

Power Management

- `command result_t powerOn();`
- `command result_t powerOff();`
- `command result_t displayOn();`
- `command result_t displayOff();`
- `command result_t backlightOn();`
- `command result_t backlightOff();`
- `command result_t sleep();`
- `command result_t sleepWakeUp();`
- `command result_t standBy();`
- `command result_t standByWakeUp();`

Table 1: SeeMote driver low level API

3.2. User Interface

The driver API is listed in the Table 2. The LCD driver supports the following groups of functions: (a) graphics modes, such as setting the color, (b) drawing graphical primitives such as pixels, lines, rectangles, and filled boxes, (c) text output such as a character, a string, and an integer in decimal or hexadecimal formats, and (d) support for user interface elements such as menus.

Graphics Output

```
• uint16_t RGB16(red, green, blue);
• command uint16_t setColor( uint16_t c );
• command uint16_t setBgColor( uint16_t c );
• command void clear(LcdColor_t color);

• command void pixel( uint8_t x, uint8_t y);
• command void line( uint8_t x1, uint8_t y1,
      uint8_t x2, uint8_t y2 );
• command void rect( uint8_t x, uint8_t y,
      uint8_t w, uint8_t h);
• command void box ( uint8_t x, uint8_t y,
      uint8_t w, uint8_t h);
```

Text Output

```
• command void gotoXY(uint8_t x, uint8_t y);
• command void nl();
• command void chr( char ch );
• command void print( char *textPtr );

• command void printInt8( int8_t val );
• command void printInt16( int16_t val );
• command void printInt32( int32_t val );
• command void printHex8( uint8_t val );
• command void printHex16( uint16_t val );
• command void printHex32( uint32_t val );
```

User Interface

```
• command uint8_t menu( uint8_t numItems,
      char** labels );
```

Table 2: SeeMote driver high level API

The driver and a small test application take up about 8KB of the program memory and 2.5KB of the random access memory (RAM). The cache memory is primarily used for generating the graphical text representation prior to copying it to the screen. The footprint in RAM is mostly due to the cache memory and could be reduced by 2KB at the expense of the text drawing speed.

3.3. Extended Functionality

Currently, we are working on extended functionality such as displaying a bitmap and a screen capture. However, the screen capture functionality is limited by the RAM size of the MICAz mote. The screen data has to be transferred directly from the video RAM to the radio packet by packet, while inhibiting any output to the screen until the capture is done. Applying data compression techniques on the fly may reduce the traffic somewhat. However, the priority for the screen capture functionality is low because of the lack of applications. Most likely it will be used for capturing screen images for the software documentation. A good digital camera could do the same.

More interesting functionality is to design a protocol for a remote display unit to which other WSN nodes can connect over a wireless link and display graphical primitives or text. This functionality is essential for remote debugging.

4. APPLICATIONS

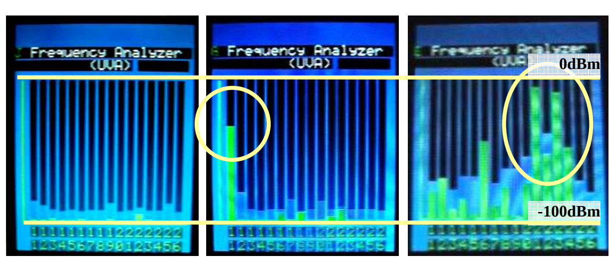
In this section we present several applications developed at our lab and discuss other potential uses of the SeeMote.

4.1. Pocket Frequency Analyzer

The current MICAz and Telos motes adopt the CC2420 radio, which conforms to the IEEE 802.15.4 standard [13]. This radio transceiver works on the 2.4 GHz ISM band and is using 16 channels, 5-MHz wide each [14]. With the growing deployment of wireless sensor networks, radio interference exists both within a network and between neighboring networks. The radio communication also suffers interference from other electronic devices, such as wireless keyboards and mice, wireless remote controllers, cell phones and even microwave ovens.

All these devices add to the difficulty to test and debug wireless communication protocols, since it is hard to locate the source of noise. Plus, sensor devices within a home-care network should avoiding frequencies that can be interfered by the microwave oven in the kitchen. Otherwise, the sensor network may perform intermittently, working correctly only when the microwave is turned off.

A frequency analyzer is a perfect tool for resolving these problems. It can monitor the interference signal power level, generated by existing electronic devices, whether they are sensor devices or not, within the deployment environment. When the background noise levels within the 16 channels are collected, the newly introduced devices can be configured to use the channels that are the least affected by the interference. Since a commercial frequency analyzer is rather large in size, power hungry, and expensive, a pocket frequency analyzer with the presented LCD module is a much better choice.



(a) (b) (c)

Figure 3: Frequency Analyzer display in different environments

Figure 3 shows three screenshots of the frequency analyzer application running in different environments. The display indicates sixteen bars that represent the runtime channel loads of the respective channels in the 2.4 GHz band, measured in dBm. Figure 3 (a) plots the power level when there is no significant radio interference present. We can see that the power level readings of all channels are small, close to -100dBm, showing that the channels are idle. When an existing sensor network, working on channel 11, is turned on, Figure 3 (b) demonstrates that the power level reading in channel 11 increases. Also, the neighboring channel, channel 12, is affected. Figure 3 (c) illustrates the case when the microwave oven in the kitchen is turned on. It is shown that channels 21 to 24 are greatly affected by the microwave radiation and should be avoided being used by any home-care sensor networks.

4.2. In-Situ/Remote Sensory Data Display

The LCD module enabled nodes are capable of immediate sensory data display while in the field. We created an application RfmToLcd that is similar to the application RfmToLeds included with the TinyOS distribution. Our application receives data over the radio and plots them on the LCD screen. Figure 4 shows the application plotting an ECG diagram that has been transmitted from a remote node. Virtually any sensory data can be plotted by the application.

Figure 4: Remote sensory data display

In addition, the application has been a valuable debugging tool. It is a challenge to pin down errors in the program that is running on a mote with no visual feedback. Often programmers use the three LEDs built in the traditional motes such as MicaZ or Telos Sky to display the variable values or current ADC readings on a remote node. However, it is difficult to monitor the actual data that can assume more than 8 states. The LCD mote can show detailed information on the screen. The code to be debugged can be run on the LCD node itself, or on a remote node connected to the LCD mote over a RF or serial link, if the application requires the use of another sensor board. The programmer can easily move between the deployed nodes and query them to find weather the distributed system is functioning properly.

4.3. Other Applications

The value of the multimodal user interface module (LCD mote) is the symbiosis of the components that, used in combination, enable a variety of applications as discussed below (Figure 5).

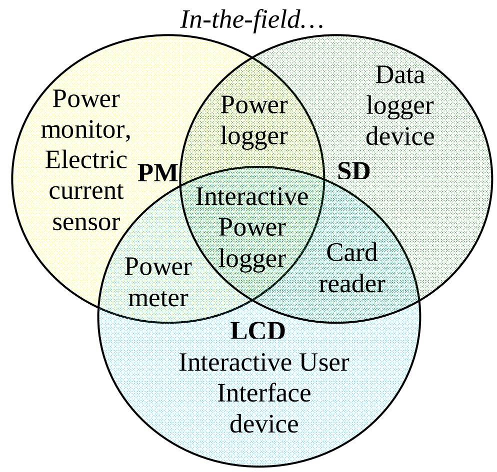


Figure 5: Application space for the LCD, Power Meter (PM) and SD/MMC card logger (SD)

The LCD display and the user navigation buttons enable applications such as monitoring and configuration of the WSN as the user moves through the area where the network is deployed, i.e. in the field. Consider an environmental scientist who is in the middle of the WSN that is deployed in the forest. The scientist may need to observe or recalibrate the sensors as he sees fit while observing the surrounding environment. He may query the closest node for the sensory readings, or ask for the ambient temperature to a group of nodes. He may also store the query results directly to the removable storage card and analyze the data using a more powerful computing platform later.

In another application the person responsible for the maintenance of the WSN may approach and test the WSN nodes that appear faulty. The nodes may indeed be faulty, or just have their power supply depleted and therefore unable to communicate over larger distances. The LCD screen can show the radio communications power as the user moves through the field, and map the radio topology, which is not uniform in realistic situations [15]. In addition, the user can debug the network while in the field by monitoring the wireless sensor network traffic and performance statistics.

Yet another application is mounting a mote with our LCD module on customized glasses and delivering the image from the LCD to the user's eye optically through a beam splitter. This enables the user to request reference information in his or her field of vision without diverting his line of view and without impairing his mobility. For example, a surgeon may request the patient's history report or an x-ray while performing the surgery. Alternatively, an electronics engineer may request a datasheet of a chip while he is testing a circuit on his or her workbench.

The power-meter capability of the LCD mote enables the user to measure the actual power consumption of external devices. This can be done on both a local and global scale. On the local scale a single WSN node is instrumented with the LCD mote and observed for its power consumption characteristics. Suppose a researcher desires to profile the power requirements for an algorithm. He can start the power measurements at the beginning of the algorithm, and then stop after the algorithm is finished. The power-meter logs the power profile. Thus, different algorithms can be evaluated with respect to their power consumption. This is also possible on a global scale, by instrumenting a group of network nodes with the power-meter nodes. The measurements are logged on the removable memory cards. Thus, the power-meter motes do not need to use the radio communications during the experiment and will not have impact on the experiment results. Also, by using the removable storage the system is not limited to the built-in 1MB flash memory chip, as the current SD memory cards are capable of storing gigabytes of information.

5. COMPARISON

One might argue that many of the applications discussed above and most of the functionality provided by the SeeMote are already enabled by using a laptop or a PDA. In many cases it is true; however one must consider the drawbacks of these solutions: the power consumption of such devices is much higher. A laptop runs for about 4 to 6 hours before its batteries are depleted, which is the price of the higher performance. It is also easier to carry a small and light wearable mote than a relatively big and bulky laptop. A PDA, although smaller and less power hungry than a laptop, is still heavier than a mote with the SeeMote attached and has a limited time of operation due to the higher computational power it provides. In either case there is also a need for a wireless gateway device between the laptop or PDA and the WSN, which incurs additional weight, power consumption, and cost.

The LCD module, however, is a small, self-contained device that provides sufficient functionality for the user interface and naturally interfaces with the host mote via the standard 51-pin connector. The power consumption of the LCD module running the remote sensory data display application with the display on is 60mA to 110mA at 3V depending on the backlight intensity, which translates to 180mW to 330mW (Table 3). An iPaq PDA running Microsoft Windows CE operating system consumes 1.72W to 2.8W depending on the backlight brightness level [16]. Power consumption for laptops is even higher [17]. This illustrates that the SeeMote module is a good choice for a power-conscious user.

	Power (with backlight, mW):		**Resolu -tion**	**Weigth w/o batteries**	**Approx. price**
	Low	**High**	**(pixels)**	**(g)**	**(USD)**
MICAz	90-100		NA	~18	125
MICAz+ SeeMote	180	330	128x 160	~40	170
iPAQ PDA	1720	2800	240x 320	~190	300
Laptop	6000	14000	1024x 768	1000-3000	500-2000

Table 3: User interface node comparison. MICAz is running In-Situ/Sensory Data Display application under TinyOS with the radio reception enabled.

It is easy to notice that the SeeMote is also much lighter in weight and smaller in dimensions than a PDA. We estimate that the SeeMote could be manufactured for approximately 45 USD at large quantities, the LCD screen with the integrated controller being the most expensive part. Even together with the MICAz mote the cost is less than a PDA. The only category where the SeeMote is losing to PDAs and laptops is the screen resolution, although it is not that far from the PDAs. However, considering the size of the SeeMote and our experience with the applications, the resolution is quite sufficient for displaying the up to date information of interest.

6. CONCLUSIONS

We have designed a SeeMote sensor module that provides a graphical in-the-field interface for wireless sensor networks. The module is fully implemented as a sensor-board for MICAz motes. The sensor module size is 34x58x12mm. When attached to the MICAz mote the SeeMote increases the height by 10mm while not increasing the length or the width. It has a color LCD display and five way navigation buttons for user interface. The module has a MMC and SD memory card interface that allows for removable storage. Finally, the module has a power meter component that samples external power supply current and voltage, thus enabling the power consumption monitoring for an external device.

The SeeMote module enables a set of applications for WSN diagnostics, configuration, and debugging, for example, a Pocket Frequency Analyzer, an In-Situ/Remote Sensory Data Display. The SeeMote in concert with such applications is better used for certain types of large area, hard to reach, and wearable WSN, where use of laptops and PDAs is prohibitively inconvenient or expensive.

For the future we are considering several directions. The sample rate of the Atmega128 ADC is rather low for fine grain power measurements. Therefore, we are considering designing a power meter with external high-speed ADC. Another direction is to create a fully functional miniature network node with a microcontroller included on a single board. Such module would be able to interface with the host mote, while unaffecting the processing and I/O resources of the host mote. It would provide more powerful debugging features, for example, debugging a mote with a sensor board attached, and also displaying sensory data or incoming wireless traffic at higher rates.

REFERENCES

[1] G. Werner-Allen, J. Johnson, M. Ruiz, J. Lees, M. Welsh, "Monitoring Volcanic Eruptions with a Wireless Sensor Network," in 2nd European Workshop on Wireless Sensor Networks, Istanbul, January 2005.

[2] G. 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 Proc. SenSys, November 2005.

[3] B. Lo, G. Z. Yang, "Key Technical Challenges and Current Implementations of Body Sensor Networks," Proc. 2nd International Workshop on Body Sensor Networks (BSN 2005), April 2005.

[4] D. Malan, T. Fulford-Jones, M. Welsh, S. Moulton, "CodeBlue: An Ad Hoc Sensor Network Infrastructure for Emergency Medical Care," International Workshop on Wearable and Implantable Body Sensor Networks, April 2004.

[5] W. Brunette, J. Lester, A. Rea, G. Borriello, "Some Sensor Network Elements for Ubiquitous Computing," Information Processing in Sensor Networks, IPSN 2005, Fourth International Symposium on 15 April 2005 Page(s):388 – 392.

[6] Crossbow Technology Inc., "MICAz Wireless Measurement System," [Online Document], Document Part Number: 6020-0060-01 Rev A, http://www.xbow.com/

[7] Lightsonic Optoelectronic Inc., "1.79" 65K TFT LCD Module Preliminary Specification, Model Name: LTM018A600," DOC. NO.: TA-LTM018A-6000 VERSION: A.2, [Online Document], http://www.loi.com.tw/

[8] Linear Technology Corporation, "LT1937 - White LED Step-Up Converter in SC70 and ThinSOT," [Online Document], http://www.linear.com/

[9] Himax Technologies Inc., "HX8302A Data Sheet 396-CH, 65536-Color TFT Source Driver with Internal RAM," [Online Document], http://www.himax.com.tw/

[10] Secure Digital Association, "SD Input/Output (SDIO) Card Specification," [Online Document], Version 1.00 October, 2001, http://www.sdcard.org/

[11] MMCA Technical Committee, "The MultiMediaCard, Based On System Specification Version 4.1, System Summary," [Online Document], http://www.mmca.org/

[12] Atmel Corporation, "8-bit AVR Microcontroller with 128K Bytes In-System Programmable Flash," [Online Document], 2004, http://www.atmel.com/

[13] Institute of Electrical and Electronics Engineers, Inc., "IEEE Std 802.15.4™-2003," The [Online Document], October 2003, http://standards.ieee.org/

[14] Chipcon Inc., "CC2420 - 2.4 GHz IEEE 802.15.4 / ZigBee-ready RF Transceiver," [Online Document], ver. 1.3, http://www.chipcon.com/

[15] G. Zhou, T. He, S. Krishnamurthy, and J.A. Stankovic, "Impact of Radio Irregularity on Wireless Sensor Networks," ACM MOBISYS 2004, Boston, Massachusetts, June 2004.

[16] S. Pasricha, S. Mohapatra, M. Luthra, N. Dutt & N. Venkatasubramanian, "Reducing Backlight Power Consumption for Streaming Video Applications on Mobile Handheld Devices," ESTIMedia 2003, First Workshop on Embedded Systems for Real-Time Multimedia, October 2003.

[17] L.M. Feeney, M. Nilsson, "Investigating the Energy Consumption of a Wireless Network Interface in an Ad Hoc Networking," IEEE INFOCOM, 2001.