MicroLEAP: Energy-aware Wireless Sensor Platform for Biomedical Sensing Applications

Lawrence K. Au, Winston H. Wu, Maxim A. Batalin, Dustin H. McIntire and William J. Kaiser

Electrical Engineering Department
University of California, Los Angeles
Los Angeles, CA 90095
Email: au@ee.ucla.edu

Abstract—Extended system lifetime is a critical requirement for wearable sensor platforms. However, these platforms must also accommodate local data processing, data storage, and broadband wireless communications. Since compact battery storage capacity is constrained, there exists a fundamental tradeoff between energy optimization and performance. Furthermore, biomedical transducers may also demand high peak power dissipation during active operations. Energy management, therefore, must be introduced through new hardware architecture and enabled through software in the overall system design. To effectively optimize energy dissipation for biomedical sensing applications, a new wearable sensor platform, MicroLEAP, has been developed. The MicroLEAP platform supports per-task real-time energy profiling to permit adaptive applications that select platform components to best match dynamically-varying measurement requirements. MicroLEAP design, implementation, and example of energy-aware operation are demonstrated.

I. INTRODUCTION

Biomedical wearable sensor platforms often deal with conflicting requirements: support of complex sensing, computing, communication functionality, limited battery capacity, and compact geometry for wearability [1]. They must be available in lightweight and unobtrusive packaging in order to be worn comfortably by human subjects. In addition to data storage and broadband communication requirements, these platforms must also provide adequate sampling resolution and sampling rate to ensure that signal processing algorithms can be performed reliably and accurately. Energy resources, on the other hand, are limited since most wearable platforms are battery-powered. It is important therefore to strike a balance between sensing requirements and severe energy constraint.

Wireless Body Sensor Networks (BSNs) have largely focused on architectures based on sensor “motes”, which are usually equipped with embedded processors, low-power radios, and physiological sensors. The CodeBlue project from Harvard University uses Crossbow’s MICA2 to implement a wearable platform with many important capabilities [2]. Jovanov et al. have implemented an electrocardiogram (ECG) sensor board, oximeter sensor, and other signal processing modules on the Moteiv’s Tmote Sky platform [3]. The BTnode provides a dual-radio (CC1000 and Bluetooth) architecture for potential BSN applications [4]. The iMote provides an ARM7-based embedded processor and a Bluetooth radio [5]. All of these research platforms are well-supported through open-source communities, facilitating hardware development efforts. There is also existing open-source software (e.g. TinyOS [6]) that allows rapid algorithm implementation on such platforms.

Existing sensor motes, however, have several shortcomings when used for biomedical sensing applications. First of all, the resolution of the analog-digital converters (ADCs) provided on motes are typically either 10-bit or 12-bit. In contrast, sampling of physiological signals such as ECG requires a sample resolution of 16 bits or higher [7]. Data storage capacity for sensor data on these platforms is also inadequate for operations on the sensor platform over an extended period of time during which wireless communication may not be available. In regards to radio communications, sensor motes are designed for long-term, low data-rate applications such as environmental sensing, where the sensor sampling rate (e.g. on the order of 1 Hz or lower) and transmission bandwidth requirements are low. In biomedical monitoring, however, data are often collected at hundreds of hertz with 16-bit resolution or higher. Additionally, the high sensor data sampling rate often requires high communication rate and reliable transmission.

An urgent requirement for wireless BSN platforms is that of energy-aware operations to accommodate the simultaneous demands of performance and long lifetime. This requires that energy dissipation sensing be available within the platform itself in order to allow proper adaptation. Energy models of sensor motes have been studied in computer simulations. For instance, PowerTOSSIM can model the energy consumption of static TinyOS applications on MICA2 motes [8]. Most of the offline simulations, however, may provide variable or inconsistent results on adaptive algorithms, since the system’s behavior is dependent upon real-time data (e.g. temperature, light intensity) from the physical environment and from end-users, both of which cannot be reliably predicted. In the context of biomedical sensing, phenomena are often dynamically changing. As a result, the ability to perform energy profiling in real time and adapt the operations of an algorithm accordingly is particularly beneficial.

The MicroLEAP (µLEAP) platform provides a new architectural solution for BSN applications, addressing the limitations of past platforms and introducing real-time energy profiling. Its design emphasizes the unique requirements in
BSNs, differing from those for environmental monitoring that have driven the LEAP platform [9]. It provides the required sensing resolution and a radio interface for compatibility with the consumer wireless electronic devices with which BSN systems must be interoperable. Most importantly, MicroLEAP provides real-time energy profiling and management on a wearable sensor platform through algorithms that access a convenient software interface. Signal processing algorithms and broadband communications can be performed on demand, and corresponding energy consumption data can be captured in real time. This hardware-software system approach also encourages the usage and scheduled operation of energy-efficient components, a key consideration during system design. The distributed nature of BSNs also allows multiple very low-power sensors to collectively infer patient context through data fusion (as in the MEDIC system [10]), an opportunity for further reduction in energy dissipation. Additionally, the proposed sensor platform can also be generalized to other potential applications such as acoustic sensing, aquatic sensing, structural monitoring, and other applications where adaptive sensing is employed.

II. DESIGN CONSIDERATIONS

To support the functionalities mentioned in Section I, a wearable sensor platform should include a low-power embedded processor for basic data processing, a high-resolution ADC, data storage, high-performance radio, sensors, as well as energy profiling and management support. From a software design perspective, the system must be equipped with a real-time operating system in order to provide accurate energy profiling along with multitasking capability for data processing, data transmission, and sensing. The platform should also be modular, allowing users to rapidly deploy new types of sensors and radios for various BSN designs.

A. Embedded Processor

For wearable platforms, the processor should provide an instruction set that can efficiently implement basic signal processing algorithms (e.g. Fast Fourier Transforms). Furthermore, it should contain multiple low-power operating modes for energy-aware scheduling. Embedded processors such as Texas Instruments’s MSP430 and Atmel’s AVR are viable options. MicroLEAP uses MSP430 due to its 16-bit RISC architecture and extremely low-power operating modes. Atmel’s 8-bit processor has very low power consumption, but typically consumes more energy due to longer execution time [11].

B. Analog-Digital Converter (ADC)

As noted above, biomedical signals should be digitized with 16-bit resolution or higher. Because the internal ADCs on embedded processors are usually 12-bit or lower, an external high-resolution ADC with multiple channels is required to interface with multiple sensors. MicroLEAP uses an 8-channel, 16-bit external ADC from Texas Instruments.

C. Data Storage

Additional data storage such as flash memory provides users the capability to explore the energy tradeoff between real-time data streaming and local data processing. For instance, signal processing algorithms on the platform can constantly analyze the acquired sensor data and transmit them through the radio in a delay-tolerant fashion. This intermittent use of the radio may reduce the overall energy consumption. An 8-Mbit flash memory is chosen due to its storage capacity and its small package size: assuming no additional storage overhead, an 8-Mbit flash can accommodate approximately 34 minutes of ECG waveforms generated by a 2-lead ECG circuit on one sensor channel at 250 Hz [7].

D. Radio Communication

IEEE 802.15.4 has been promoted as the main wireless communication channel in BSNs research. In this paper, however, we propose using Bluetooth. The reasons are as follows. Firstly, most portable electronic devices such as cell phones and PDAs are equipped with Bluetooth. Integrating biomedical sensing capabilities into commercial portable devices with 802.15.4 implies major overhaul of the current Bluetooth-based Personal Area Networks. Secondly, compared to environmental variables such as ambient temperature, physiological signals such as ECG require relatively high data rates for any signal processing algorithms to capture useful information. With high packet loss rates and potential wireless interferences in real-life scenarios, data streaming through 802.15.4 becomes a major concern [12].

Bluetooth also offers potential efficiency in terms of the energy required to transmit a fixed number of data bits. Take Chipcon’s CC2420 transceiver as an example of an 802.15.4 radio: with an output power of 0 dBm, it consumes about 57.4 mW at maximum data rate of 250 kbps [13]. A Class-2 Bluetooth 2.0-enabled module (which has an output power of 4 dBm) consumes 103.9 mW at 600 kbps (as supported by the protocol stack) [14]. Table I illustrates the values. If the Bluetooth module can be duty-cycled with energy-aware scheduling algorithms (as opposed to remain on at all times), the overall energy consumption will be lower. The time required for transmission is also shorter for Bluetooth due to its higher data rate.

New radio standards may also be incorporated in the near future. For instance, the Bluetooth Ultra Low Power (ULP) is expected to consume 10% to 25% of the power of normal Bluetooth operations. Ultra-wideband (UWB) radio has also

<table>
<thead>
<tr>
<th>Power (mW)</th>
<th>Data Rate (kbps)</th>
<th>Energy/Bit (nJ/bit)</th>
</tr>
</thead>
<tbody>
<tr>
<td>IEEE 802.15.4 Bluetooth</td>
<td>57.4</td>
<td>250</td>
</tr>
</tbody>
</table>

TABLE I 802.15.4 VS. BLUETOOTH - TRANSMITTING AT MAXIMUM DATA RATE [13], [14]
Fig. 1. MicroLEAP Architecture

E. Energy Profiling

In order to incorporate energy-aware algorithms, the underlying hardware architecture of a typical sensor platform must be modified. Major components on the proposed system should be equipped with appropriate current-sensing circuitry. The hardware-software system approach is especially critical since the capabilities provided in the circuitry directly translate to how effectively energy can be monitored and managed. At the software level, a software-controlled enabling scheme must also be available to power on or off components. This allows developers to control what and when to perform energy-aware algorithms. The internal ADC in the processor selected for MicroLEAP provides sufficient support for energy profiling; thus, an additional ADC is not required. The interrupt capability provided with the internal ADC also facilitates per-task energy profiling.

III. HARDWARE ARCHITECTURE

The MicroLEAP platform consists of three different system boards: Processor Board, Radio Board, and Sensor Board. The simplified architecture is shown in Fig. 1.

A. Processor Board

The Processor Board contains the MSP430 embedded processor, an external 16-bit ADC, 8Mbit of flash memory, and the respective current sensing circuitry (Fig. 2) for energy management. MSP430 provides multiple low-power modes to reduce power consumption during idle periods. The external 16-bit ADC supports an effective data sampling throughput of 100 ksps, sufficient for typical sampling requirements. Different communication ports are available for communications between components.

1) Energy Management Unit: The Energy Management Unit (EMU) is responsible for measuring energy consumption and power-cycling components. Six components are monitored: processor, radio, external flash, sensors powered by 3.3V voltage supply, sensors power by 5V voltage supply, and the external ADC. To enable the ability to monitor energy consumption at a fine-grained, per-task level, a current-sensing circuitry is employed (Fig. 2) to capture the current consumption characteristics of each component. A current-sense resistor, $R_{\text{sense}}$, is placed in series with each regulated voltage supply. The sensed voltage is amplified and periodically sampled by the internal ADC on the embedded processor.

The energy sample generated by the internal ADC during each instance is added to the previous sum. From each of the cumulative sums of the components, corresponding charge and energy values can be calculated using Equations 1 and 2.

$$Q = \frac{V_{\text{ADC,range}} \cdot \text{sum}}{(2^B - 1) \cdot A_v \cdot R_{\text{sense}} \cdot f_{\text{sample}}}$$  \hspace{1cm} (1)

$$E = Q \cdot V_{\text{supply}}$$  \hspace{1cm} (2)

where,

- $Q = \text{total charge (C)}$
- $E = \text{total energy (J)}$
- $V_{\text{ADC,range}} = \text{input dynamic range of ADC (V)}$
- $\text{sum} = \text{cumulative sum of energy samples}$
- $B = \text{number of bits available in ADC}$
- $A_v = \text{voltage gain of the amplifying stage (V/V)}$
- $R_{\text{sense}} = \text{current-sense resistor (Ω)}$
- $V_{\text{supply}} = \text{supply voltage of the component (V)}$
- $f_{\text{sample}} = \text{sampling frequency of EMU (Hz)}$

The sampling frequency of the EMU depends on the application. A low sampling frequency may neglect components that potentially have high-frequency power switching, whereas a high sampling frequency introduces higher sampling overhead.

B. Radio Board

The Radio Board has a detachable design and is equipped with a Class-2 Bluetooth module [14]. The protocol stack of the Bluetooth radio consists of an application layer which allows direct control through serial communication.

C. Sensor Board

The Sensor Board consists of a 3-D accelerometer [15] and two ECG circuits [16]. Appropriate amplifying circuits are embedded to match the input dynamic range of the external ADC.
IV. SOFTWARE ARCHITECTURE

MicroLEAP uses μC/OS-II, a multitasking, preemptive real-time operating system, due to its small kernel footprint and relaxed constraints on hardware [17]. It contains all primary operating system features such as a scheduler, semaphores/mutexes, memory management, and task synchronization. Device drivers and user-level APIs are designed in a layered approach so as to abstract any hardware details when developing new applications.

To determine per-task energy consumption, the internal ADC can be enabled at the beginning of a particular task and disabled whenever a context switch occurs. When a task is interrupted, the corresponding Interrupt Service Routine (ISR) may temporarily disable the EMU and re-enable it before exiting so that the energy consumed during ISR execution (assuming significant) is not accounted into the interrupted task. The additional overhead of enabling or disabling the EMU is also minimal since each operation requires one machine instruction. A conceptual diagram of a per-task energy profiling example is shown in Fig. 3: the OS scheduling algorithm triggers a context switch to a new task (1), and this current task activates EMU (2). An interrupt is triggered, so the EMU is temporarily disabled (3). ISR executes its instructions and re-enables EMU (4) before returning. The task resumes and disables EMU before another context switch occurs (5 and 6).

To demonstrate the energy profiling capability, a two-task application is developed on the MicroLEAP platform. SensingTask() collects both sensor data and energy profiles at predetermined frequencies and transmits them immediately to a remote client through Bluetooth. QueryTask() is responsible for polling new configuration commands from the remote client. Sampling frequencies of both data and energy sensing are configured through configuration commands. To access the sensor and energy data, the remote client executes a program that includes a Bluetooth library for the Bluetooth radio on the respective system. For instance, the Linux operating system provides an open-source Bluetooth protocol stack that allows transmitting and receiving data packets through basic socket programming [18]. Data parsing and processing can be performed on received data.

V. EXPERIMENTAL RESULT

A photo of MicroLEAP is shown in Fig. 4.
application development that may benefit from the novel capabilities of MicroLEAP. A simple low-power scheme can reduce the power consumption to one-tenth of that in active mode, suggesting that components such as radio should be used sparingly (i.e. only when necessary).

Future work includes deployments of MicroLEAP in biomedical sensing applications such as MEDIC [10], [19]. A more thorough study on how to utilize the energy information is critical, and energy-aware algorithms are under development that balance the tradeoff between diagnostic inference certainty and energy consumption. Local processing techniques may also be developed to further reduce energy consumption.

REFERENCES