IntellBatt: The Smart Battery

Suman Kalyan Mandal
Texas A&M University
skmandal@cs.tamu.edu

Praveen S. Bhojwani
Sun Microsystems
praveen.bhojwani@sun.com

Saraju P. Mohanty
University of North Texas
saraju.mohanty@unt.edu

Rabi N. Mahapatra
Texas A&M University
rabi@cs.tamu.edu

IntellBatt: The Smart Battery

Suman K. Mandal, Praveen S. Bhojwani, Saraju P. Mohanty, and Rabi N. Mahapatra

ABSTRACT

This article introduces IntellBatt; a novel design of a multi-cell battery. IntellBatt exploits the cell characteristics to enhance battery lifetime, to ensure safe operation and to deliver better performance. Experiments were performed using Li ion cells and a portable DVD player. Simulation of the obtained current trace with IntellBatt have shown an enhancement of battery lifetime by 22%.

POWERING INTELLIGENTLY

The predominance of battery-operated, wireless handheld devices motivates the efficient and robust use of a limited energy supply. Managing battery life and safety of the system are critical design constraints for portable systems. Battery-Aware Task Scheduling (BATS) techniques have been used to ensure longer usage of a battery, while considering discharge/recharge characteristics. Smart batteries to manage the safety of battery packs have been deployed to manage concerns about high temperature, over-discharge, and overcharge (see the sidebar).

A paradigm shift to these traditional approaches would involve deploying a smarter battery into the system. Such an intelligent battery would manage the discharge or recharge of the cells while ensuring battery life. Battery safety is also managed to ensure robustness and reliability. Toward this direction, this article presents **IntellBatt**, an *intelligent battery cell array* (IBCA) design. The proposed design uses multiple cells and a management unit that ensures battery life and safety. Using an IBCA organization provides the following distinct benefits: (1) closely monitors cell status; (2) dynamically selects cells to match the device requirement, reducing loss in DC-to-DC converters or similar voltage and/or power regulators; (3) exploits charge recovery effect in cells, thereby giving advantages over monolithic batteries; and (4) provides the possibility of preprogramming the discharge pattern if the current profile is stable and known a priori. IntellBatt assumes no knowledge of the device as such and hence is applicable for any battery-operated portable consumer electronics equipment. IntellBatt can operate in either a standalone fashion or can be combined with BATS techniques for maximum advantage. Simulation and experimental results presented in this article demonstrate the observed improvement.

SMART BATTERY

A smart battery is more than just a collection of cells. The most commonplace additions in a smart battery are safety circuits. More advanced smart batteries have controller circuitry that controls the charging or discharging of the battery and optionally have a communication interface, such as SMBus or I^2C, for communication with the device using the battery.

Most of today's battery packs used in laptops and portable electronics are smart batteries. IntellBatt takes smart batteries a step ahead by saving more energy and making them safer than other batteries.

IntellBatt differs from traditional smart batteries in the way it manages cells. Instead of simple connection and monitoring, it actively schedules cells to optimize capacity and charge delivery..

References

1. Smart Battery System Management Specifications; http://sbs-forum.org/specs/.
2. J. Chatzakis et al. "Designing a New Generalized Battery Management system," *IEEE Trans. Industrial Electronics,* vol. 50, no. 5, Oct. 2003, pp. 990-999.

The highlights of this article are as follows:

- The design of the proposed IntellBatt is presented, which is a novel intelligent battery cell array design that consists of multiple cells and a management unit.
- A simple and high-performance cell selection algorithm is presented that provides functionalities like cell selection, monitoring, and scheduling previously not considered.
- The proposed design is demonstrated via simulation using available accurate battery models and typical application traces and further enhancement of battery life by combining its operation with BATS approaches.

BATTERY-AWARE TASK SCHEDULING

Battery-aware task scheduling is another approach to optimize battery lifetime. In this scheme, to ensure that battery lifetimes are considered during system operation, tasks are scheduled while considering battery charge or discharge patterns. Chowdhury and Chakrabarti[1] presented a task-scheduling scheme that utilizes battery model information. These techniques attempt to tailor the current profile of the device to match the optimum discharge rate of the battery. Similarly, battery pack scheduling on the system side has been explored by Asumadu et al.[2] To contrast these existing system-side battery management techniques, the current article proposes the deployment of smarter batteries to self-manage lifetime.

References

1. P. Chowdhury and C. Chakrabarti, "Static Task-Scheduling Algorithms for Battery-Powered DVS Systems," *IEEE Trans. Very Large Scale Integration (VLSI) Systems,* vol. 13, 2005, pp. 226-237.
2. J. A. Asumadu et al., "Precision Battery Management System," *Proc. IEEE Instrumentation and Measurement Technology Conf. (IMTC)*, 2005, pp. 1317-1320.

INTELLBATT

IntellBatt is composed of three different components: *cells*, *cell switching circuit (CSA),* and *battery cell array manager*. Detailed information on their function and operation is presented in later parts of this section.

BATTERY CELL ARRAY STRUCTURE

The IBCA consists of three components. Figure 1 illustrates the envisioned system. The cells in IntellBatt are organized into banks of cells that are connected to the main terminals of the IntellBatt via a CSA. The IBCA manager manages the cells and determines their interconnection based on the required system load current. The SMBus interface[7] between IntellBatt and the system will provide for information exchange between the two entities. For simplicity, the SMBus implementation is ignored for the purpose of this research. The following sub-sections detail the structure and operation of the IBCA components.

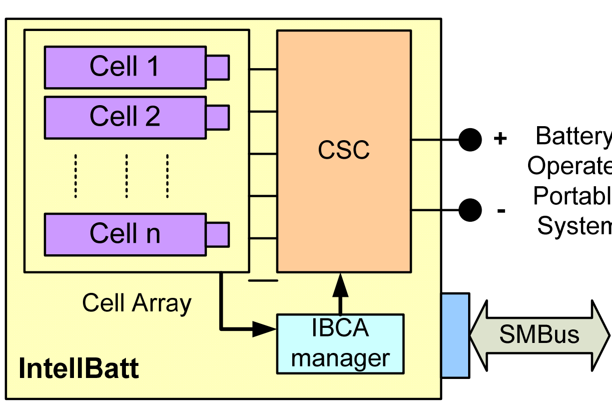

FIGURE 1: INTELLBATT STRUCTURE

INTELLBATT CELLS

The cells are organized as a collection of banks connected in series among themselves. Inside each of the banks there can be one or more cells connected in parallel to provide the required current rating from a selected bank. If the device has a rated voltage requirement V_{dev} and each cell has voltage V_{cell}, then the number of banks required is V_{dev}/V_{cell}. For applications like laptop computers, with standard V_{dev} = 10.8V ~ 11.1V and standard lithium (Li) cells with V_{cell} = 3.6V ~ 3.7V, there is a need for three banks of cells. For smaller devices like a portable DVD player or wireless media players with V_{dev} = 7.2V ~ 7.4V, the number of banks that would be needed is two. Typical battery packs uses 3 to 12 cells organized in a series or parallel combination as required. These cells provide the required load current for the target battery-operated system via the CSA.

Cell Switching Circuit (CSC)

The CSC connects cells to deliver the required current to the target system in which IntellBatt is installed. CSC configuration is performed by the IBCA manager with a code word. The CSC has to be designed to ensure the following for the IntellBatt:

- Connect cells in banks, specified via a code word from the IBCA manager to the output terminals of IntellBatt;
- Provide rapid switch reconfiguration (order of picoseconds); and
- Support the current drawn by the system without incurring significant losses in the switch.

Figure 2 shows the detail structure of the cell switching circuit. It is a matrix of two power transistor switches that can be turned on and off by a control signal. Any cell can be connected to any bank by activating the switch at the intersection of the bank and the cell.

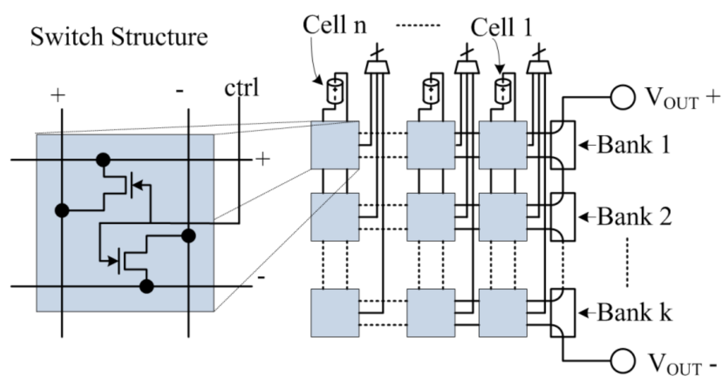

Figure 2: Cell Switching Circuit

IBCA Manager

The IBCA manager is the core of the IntellBatt system. It provides the following functionalities: monitors cell status, schedules cells for load current delivery, and ensures the safety of the IBCA.

MONITORING

The monitoring logic in the IBCA manager keeps track of the following parameters for each cell: delivery voltage, delivery current and operating temperature. These values are used to ensure safety and to make scheduling decisions. High-speed and ultra-low-power sensing circuits[9, 10] are available so that they do not increase the load on IntellBatt.

CELL SCHEDULING

IBCA Manager performs cell scheduling based on two aspects of rechargeable cells: *discharge cycle length* and total *battery life*. *Discharge cycle length* refers to the duration for which the fully charged cell can deliver the required current. This parameter is important for handheld and portable devices because the ability to run longer in a single charge cycle is desirable. *Battery life* refers to the number of discharge cycles achievable before the battery becomes unusable.

The main properties of Li ion cells that affect the lifetime and performance of the cell are *discharge current*, *discharge condition* such as temperature, *variation in the discharge current,* and *charge recovery effect*. In general, the higher the discharge current, the lower the charge delivery capability of the cell. The optimum discharge current is a property of the cell's composition. Environmental conditions like high (> 45 °C) or low temperature (< 10 °C) have been found to reduce the capacity of the cell.[8] If the discharge reaction is slower than the optimum rate (dependent on cell geometry), it results in substance buildup, which can clear if the cell is idle for a while. This effect is known as charge recovery and is exploited in IntellBatt.

Considering the aforementioned IBCA manager objectives and cell-limiting properties, the *cell selection problem* for efficient battery management can be formulated as follows:

Given I_t, the load current, and $V_{cell,i}$ for each cell, determine the best configuration of cells that meets the following criteria:

- For each cell k, $I_k < I_k^{max}$.
- For each cell k, $V_{cell,k} > V^{cuttoff}_{cell,k}$.

The minimum voltage needed in a bank is given by *Vmin*. I^{max} is the safe limiting current that can be drawn from the cell safely; i.e., not causing a short circuit. V^{cutoff} is the discharge limit beyond which the cell should not be discharged; failing to do so may lead to an adverse reaction in a cell and lead to a safety hazard. At least one cell in each bank is necessary to maintain the output voltage level. If there are **n** available cells and **k** banks, the problem can be formulated as the determination of a subset of all possible *k* clusters. The total possible solution without the constraints is $O(C_k^n * n^k)$. To limit search time in the large solution space, we use a simplified algorithm (Cell Select) to approximate the solution.

Algorithm Cell Select

```
------------------------------------------------
Input: Load Current, Individual Cell States
Output: for each cell i a bank assignment bi or 0 for idle
Initialize: for all i set bi = 0 ;
Begin
1: sort cells by voltage ;
2: for each bank
     assign one cell in descending order of voltage
   end for
3: for all banks :  cell voltage < Vmin
     while safe cells left
       add cells to match capacity in ascending order
     end while
   end for
4: for all banks
     if v(bank) is less than Vmin return failure
5: finish
End
```

Algorithm Cell Select (see sidebar) works based on the simple principle that all banks need at least one cell and the total minimum voltage of the series of banks needs to be higher than a specific minimum depending on the device that is using the battery. This internally translates into derived minimum voltage for a cell while being allocated to a bank. Algorithm Cell Select takes care of this by placing the cells in descending order of voltage to each bank one by one. This step (2: in the algorithm sidebar) ensures that maximum possible series voltage is obtained. Based on the charge state of each cell the internal resistance varies and causes voltage drop. This is reduced by step 3: in the algorithm. This allocates additional cells in a bank if its voltage is low and does so from the lowest bank voltage thus improving the neediest bank first.

The frequency of the cell selection process is governed by the chosen *scheduling scheme.* Two different scheduling schemes are possible: periodic or adaptive. In *periodic scheduling,* the cell selection operation is performed periodically independent of the change in state since the last selection operation. In *adaptive scheduling*, on the other hand, the cell selection operation interval is determined at run time.

SAFETY

Our proposed IBCA manager continuously monitors cell parameters like voltage level, temperature, and current. Unlike traditional smart batteries, which need to disconnect the power supply, the IBCA manager can shuffle the cells to control heat, current, etc. Thus, it provides right voltage and current, ensuring safe and non-interrupt operations[1].

EVALUATION

EXPERIMENTAL SETUP

IntellBatt optimizes cell utilization based entirely on knowledge of cell states and the load current. So it is necessary to obtain a realistic current drain profile using a real device for proper evaluation of the effectiveness of IntellBatt. The current profile used in the IntellBatt evaluation was obtained by using the experimental setup illustrated in Figure 3. In this case, the device is a portable DVD player, and the battery is 7.2V Li-Ion battery. The device current is redirected through a low-resistance shunt, and the voltage across the shunt is measured using a data acquisition device. The profile obtained in this experiment is used in the successive simulation of IntellBatt system as shown in Figure 3.

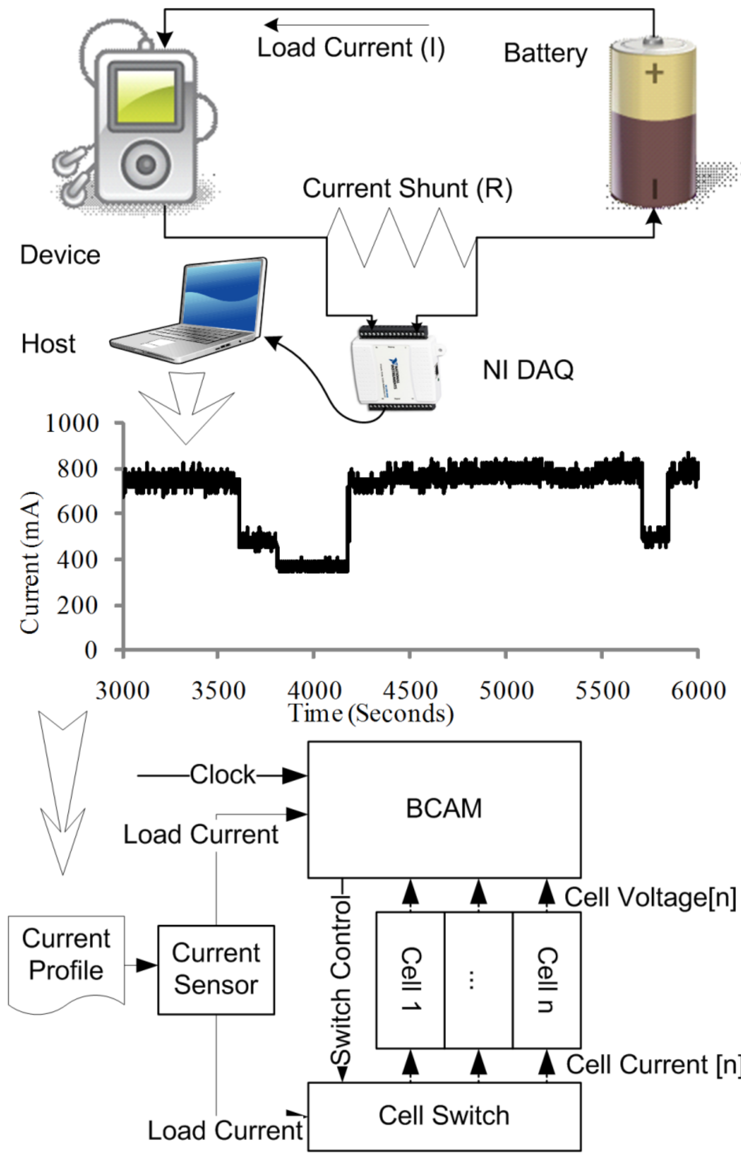

FIGURE 3: INTELLBATT EXPERIMENTAL SETUP

CELL MODEL

In the simulation environment, the cell model is used to mimic the behavior of a real cell. There are three types of cell models in the research literature: electrochemical, stochastic, and analytical. Electrochemical models are the most accurate and complex. Doyle et al. presented an electrochemical model[3] of a lithium cell, which is by far the most accurate model available.[4] Though the electrochemical model is the most accurate cell model available, it is limited by its speed. Benini et al. proposed a discrete time battery model for fast computation and quick estimation of battery lifetime.[5] More recently, Chen and Mora proposed a simpler model for lithium polymer cells.[6] This model can estimate the cell voltage with an accuracy of within 0.4 percent of Dualfoil[3] estimate.

SIMULATING THE DESIGN

The IntellBatt simulator (see Figure 3 bottom) was build using SystemC using the following components: (i) **IBCA Manager:** The IBCA manager is the main module that simulates the cell scheduling and safety algorithms. (ii) **Current Sensor**: This module feeds the recorded current profile to the IBCA manager and behaves like an analog-to-digital converter (ADC) measuring the current. (iii) **Cell Switch**: This module computes the current through each cell based on the code-word configuration input by the IBCA manager. These currents are used to update current usage in the appropriate cell modules in the design. (iv) **Cell**: Each cell module simulates a Li ion cell using the model proposed by

Chen and Mora[6] combined with the charge recovery model proposed by Rao et al.[11] This model computes the current cell voltage and writes it to the IBCA manager, hence mimicking the behavior of a real cell with voltage sensors.

EVALUATION OF THE RESULTS

Using the load current trace from a portable DVD player in the IntellBatt simulation environment, the following experiments were performed for evaluation:

- Justification for the use of multi cell batteries,
- Performance comparison against BATS
- Effect of cell scheduling scheme on IntellBatt operation
- Determination of a cell scheduling interval for IntellBatt.

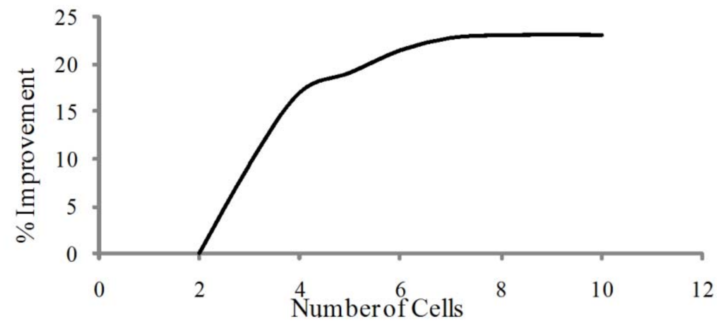

FIGURE 4: VARIATION OF DISCHARGE TIME IMPROVEMENT WITH NUMBER OF CELLS USED

IS A MULTI-CELL BATTERY BETTER?

In this experiment, the number of cells in the battery pack was increased without changing the total capacity. The use of a smaller-capacity cell improves safety,[12] and the cell is easier to manufacture. A pack of multiple batteries benefits from the recovery effect (Figure 4). In a scenario of constant load current, the benefit of adding additional cells into a multi-cell battery to exploit charge recovery effect saturates at 22 percent. The additional idle periods provided by the addition of cells begin to plateau after seven cells.

WITH BATTERY-AWARE TASK SCHEDULING

Battery Aware Task Scheduling (BATS) approaches tailor the application current profile to battery characteristics to enhance battery life. On the other hand, IntellBatt manages the cell scheduling within the battery cell array to maximize battery life. These two techniques can either be used independently or be combined for further performance improvement.

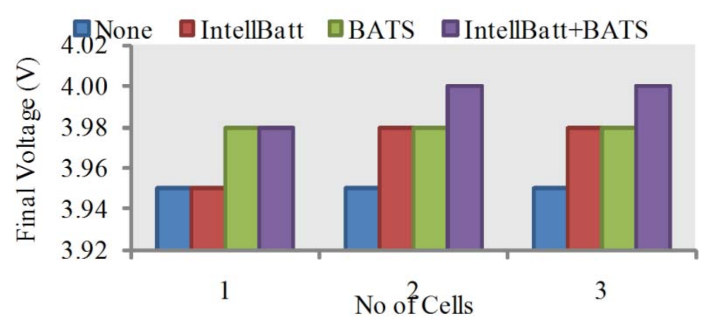

FIGURE 5: COMPARISON WITH BATTERY AWARE TASK SCHEDULING

For this comparison, a BATS algorithm illustrated in the sidebar was used. The experiment determined the final voltage available in the battery. Figure 5 compares the performance in the cases of no battery management, standalone BATS, standalone IntellBatt, and BATS + IntellBatt. In the case of a single-cell battery, BATS outperforms IntellBatt because there is no scope of cell scheduling; whereas in other cases, it performs just as well. When combined, IntellBatt compensates for scenarios when BATS is unable to tailor application current traces for battery life maximization.

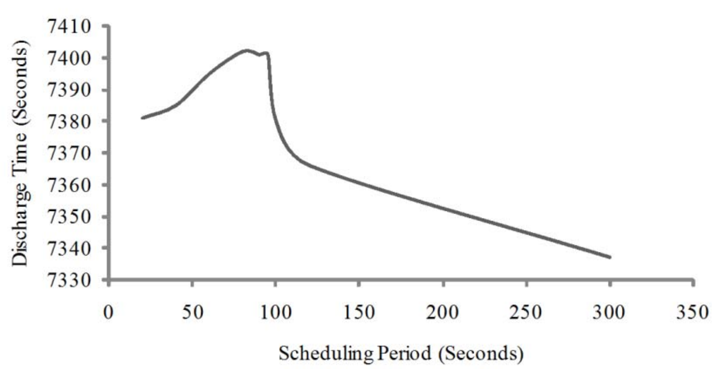

FIGURE 6: EFFECT OF SCHEDULING FREQUENCY

BEST SCHEDULING FREQUENCY

Dynamic scheduling was used with varying period to determine the effect of idle period length on charge recovery and the discharge cycle time. Too frequent or infrequent scheduling does not allow charge recovery effect to improve battery performance. Figure 6 demonstrates the effect of cell scheduling frequency on IntellBatt discharge time. For the chosen application, the ideal scheduling frequency was found to be 100 seconds.

INTELLBATT OVERHEAD

The overhead of a battery cell array based design is determined in terms of area, power, and delay. Area is not a concern since the IBCA manager and the CSC are implemented on the board level. Power and delay overheads due to the CSC are compensated by the overall benefits of the IntellBatt. The power consumed in the IBCA manager is orders of magnitude less than that of the CSC, which is of the order of microwatts.

INTELLBATT AND FUEL CELLS

Fuel cell technology is fast developing as the next-generation energy source. In this section, the applicability of IntellBatt-like designs to manage fuel cell batteries is examined.

Fuel cell-Li-ion cell analogy: Figure 7 shows the key similarity points of Li-ion cells and fuel cells. Similar to the charge recovery characteristic of Li ion cells, fuel cells respond to cell temperature. The chemical reaction in a typical fuel cell is exothermic; hence, the temperature of the cell increases with usage. With increased temperature, the efficiency of the cell goes down after a certain optimum point. In addition, smaller and less-capacity cells are easier to manufacture and are cheaper. A larger number of smaller cells can perform better than one large cell.

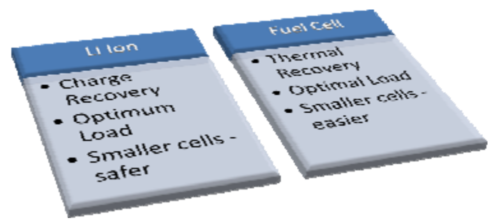

FIGURE 7: FUEL CELL VS. LI-ION CELL

Reference

1. M. Wang, H. Guo and C. Ma, "Dynamic Characteristic of a Direct Methanol Fuel Cell," *J. Fuel Cell Science & Technology*, vol. 3, May 2006, pp. 202-207.

SUMMARY AND CONCLUSIONS

In this article, IntellBatt, a novel *Intelligent Battery Cell Array* design has been proposed to offload battery management responsibilities from the system onto the battery. By using multiple cells and an intelligent battery cell array manager, IntellBatt exploits the charge recovery effect of cells combined with a cell scheduling scheme to deliver the required load to the system while enhancing battery lifetime. Besides this support, IntellBatt also addresses safety concerns due to the use of multi cell battery packs. IntellBatt can either be used standalone or can be combined with traditional battery-aware task scheduling approaches to further enhance battery life for a portable battery operated system. Our evaluation with actual current traces, obtained from a portable DVD player, demonstrated the benefit of using multiple cells and cell scheduling schemes. Future research activities in this direction will focus on a hardware prototype of IntellBatt to further validate functionality.

ACKNOWLEDGMENT

This research is supported in part by the National Science Foundation (NSF), Award Number 0509483 and Award Number 0720541.

REFERENCES

1. S.K. Mandal et al., "IntellBatt: Towards Smarter Battery," *Proc. Design Automation Conf.*, June 2008, pp. 872-877.
2. L. Benini et al., "Extending lifetime of portable systems by battery scheduling," *Proc. Design, Automation and Test in Europe*, 2001, pp. 197-203.
3. M. Doyle, T. F. Fuller, and J. Newman, "Modeling of Galvanostatic Charge and Discharge of the Lithium/Polymer/Insertion Cell," *J. Electrochemical Soc.,* vol. 140, 1993, p. 1526.
4. T.F. Fuller, M. Doyle, and J. Newman, "Simulation and Optimization of the Dual Lithium Ion Insertion Cell," *J. Electrochemical Soc.,* vol. 141, 1994, p. 1.
5. L. Benini et al., "Discrete-Time Battery Models for System-Level Low-Power Design," *IEEE Trans. Very Large Scale Integration (VLSI) Systems,* vol. 9, 2001, pp. 630-640.
6. M. Chen and G.A. Rincón-Mora, "Accurate Electrical Battery Model Capable of Predicting Runtime and IV Performance," *IEEE Trans. Energy Conversion,* vol. 21, 2006, pp. 504-511.
7. SBS Implementers Forum, "System Management Bus (SMBus) Specification," version 2, Aug. 3, 2000; http://www.smbus.org/specs/.
8. Panasonic, "Lithium Ion Batteries: Individual Data Sheet," Jan. 2007; http://www.panasonic.com/industrial/battery/oem/images/pdf/Panasonic_LiIon_CGA103450A.pdf.
9. N. Verma and A.P. Chandrakasan, "An Ultra Low Energy 12-bit Rate-Resolution Scalable SAR ADC for Wireless Sensor Nodes," *IEEE J. Solid-State Circuits,* vol. 42, 2007, pp. 1196-1205.
10. M.D. Scott, B.E. Boser, and K.S.J. Pister, "An Ultralow-Energy ADC for Smart Dust," *IEEE J. Solid-State Circuits,* vol. 38, 2003, pp. 1123-1129.
11. V. Rao, G. Singhal, A. Kumar, and N. Navet, "Battery Model for Embedded Systems," *Proc. 18th Int'l Conf. VLSI Design,* 2005, pp. 105-110.
12. R. Rao, S. Vrudhula, and N. Chang, "Battery Optimization vs Energy Optimization: Which to Choose and When," *Proc. IEEE/ACM Int'l Conf. Computer-Aided Design (ICCAD)*, 2005, pp. 439-445.

Suman K. Mandal is currently pursuing a PhD in Computer Engineering at Texas A&M University, College Station. His research interests are low-power design, multiprocessor system on a chip and network on a chip. Mandal received a B. Tech (H) in Computer Science and Engineering from the Indian Institute of Technology (Kharagpur, India). He is a student member of the IEEE. Contact him at skmandal@cse.tamu.edu.

Praveen S. Bhojwani is a member of the technical staff (hardware) at Sun Microsystems. His research interests include network on a chip, low-power design, computer architecture, and electronic design automation. Bhojwani received a PhD degree in Computer Engineering from Texas A&M University, College Station. He has been a member of the IEEE since 2008. Contact him at praveen.bhojwani@sun.com.

Saraju P. Mohanty is an assistant professor in the Department of Computer Science and Engineering at the University of North Texas. His research is in design and CAD for low-power high-performance nanoscale VLSI; he researches power, leakage, and performance models; incorporates them in CAD flow through optimization methodology; and demonstrates them through computationally intensive multimedia security applications. Mohanty received a PhD in Computer Science and Engineering from the University of South Florida in Tampa. He is a senior member of the IEEE. Contact him at saraju.mohanty@unt.edu.

Rabi N. Mahapatra is an associate professor in the Department of Computer Science at Texas A&M University, College Station, and directs the Embedded Systems and Codesign Laboratory. His research interests include the area of embedded systems, low-power design, system on a chip and VLSI design, and computer architecture. Mahapatra received a PhD in Computer Engineering from Indian Institute of Technology (Kharagpur, India). He is an associate editor of ACM Transactions on Embedded Computing, Ford Fellow and a Senior Member of the IEEE Computer Society. Contact him at rabi@cs.tamu.edu.