Battery-Driven Dynamic Power Management of Portable Systems

Luca Benini * Giuliano Castelli ‡ Alberto Macii ‡ Enrico Macii ‡ Riccardo Scarsi ‡

* Università di Bologna
Dip. di Elettronica, Informatica e Sistemistica
Bologna, ITALY 40136

‡ Politecnico di Torino
Dip. di Automatica e Informatica
Torino, ITALY 10129

Abstract

Battery life-time extension is a primary design objective for portable systems. Traditionally, battery life-time has been prolonged mainly by reducing average power consumption of system components. A careful analysis of discharge characteristics and the adoption of accurate high-level battery models in system-level design open new opportunities for life-time extension. In this paper, we introduce dynamic power management (DPM) policies specifically tailored to battery-powered systems. Battery-driven DPM strives to enhance life-time by automatically adapting discharge rate and current profiles to battery state-of-charge. The distinctive feature of these policies is the control of system operation based on the observation of battery output voltage. The effectiveness of the proposed policies and, more in general, of the idea of accounting for battery behavior during system design, is proved by the experiments carried out on a realistic case study, namely, an MP3 audio player.

1 Introduction

The activity of several components in a computing system is event-driven; for example, the activity of display servers, communication interfaces, and user interface functions is triggered by external events and it is often interleaved with long, idle periods. An intuitive way of reducing the *average power* dissipated by the whole system consists of shutting down the resources during their periods of inactivity. In other words, one can adopt a dynamic power management (DPM) policy that dictates how and when the various components should be shut down according to the system *workload*.

Workload-driven DPM has shown to be extremely effective, thanks to sophisticated policies, based on complex computational models (e.g., Markov chains) proposed in the recent literature (see [1, 2] for a complete survey).

We observe, however, that minimum average power is not always the objective when designing battery-operated, mobile applications. Rather, what really matters for this kind of systems is ensuring long battery life-time.

It has been shown in [3] that average power reduction and battery life-time extension may be numerically far apart. This implies that optimizations for minimum average power may not be equally effective in extending battery life-time, and vice versa.

Being the state-of-charge (SOC) of the battery what really needs to be preserved during system operation, taking it into account while managing the activity of all system components seems an obvious, yet necessary constraint. In this paper, we thus propose several DPM policies specifically tailored to battery life-time maximization. In particular, we introduce the class of *closed-loop* policies, whose decision rules used to control the state of operation of the system are based on the observation of battery's output voltage (which is related, non-linearly, with the SOC). This is in contrast with *open-loop* (i.e., workload-driven) solutions, that take decisions about component shut-down independently from battery voltage measurement.

Open-loop policies are normally simpler, but less effective, than closed-loop ones; therefore, they are the only viable option when cost constraints prevent the use of a voltage sensor on the battery terminals. On the other hand, the distinguishing feature of closed-loop policies is that they control system operation based on the observation of both system workload and battery output voltage. As a consequence, they can dynamically adapt the components shutdown scheme to the actual state-of-charge of the battery.

Key for the study and development of battery-driven DPM policies is the availability of a discrete-time model of the battery subsystem (i.e., battery cell and DC-DC conversion circuit) that enables the simulation of the complete application for realistic periods of time [3].

We first discuss policies for the simple case of single-battery systems; then, we move to the more attractive, yet more complicated case of multi-battery systems. Here, the ability of monitoring the behavior of the battery during system operation is coupled with the chemical capability of a battery cell of recovering some of the charge it can deliver if some resting is allowed after a current load is sustained for a sufficiently long period of time.

All closed-loop policies have been developed and tested on a realistic case study, namely, an MPEG 2-Layer 3 (MP3) digital audio player. The results we present, although preliminary, are very promising and clearly indicate that battery-driven dynamic power management, possibly combined with open-loop policies, constitute a viable solution to achieve significant battery life-time extension in portable, battery-operated applications.

The rest of the paper is organized as follows. Section 2 summarizes the discrete-time battery model of [3]. Section 3 first provides a block-level description of the MP3 player we have used as our case study; then, it introduces closed-loop policies for both single and multi-battery systems. Section 4 presents results on the usage of the new policies and compares them to those obtained with open-loop techniques. Finally, Section 5 concludes the paper.

2 System-Level Battery Model

The system-level, discrete-time model of a Lithium-Ion battery that we have introduced in [3] is derived from the circuit-level continuous-time model originally proposed in [4, 5].

Charge storage in a battery can be modeled as a capacitor with capacitance $C = 3600 \cdot CAP$, where CAP is the nominal capacity in Ahr, which is usually provided in the battery's data-sheet. By setting the initial voltage across the capacitor $V_C = 1$, we initialize the battery to its fully charged state. Unfortunately, the simple linear capacitor model is not accurate enough to model complex phenomena observed during battery discharge. In fact, the following three major effects must be taken into account:

- Battery voltage depends non-linearly on its SOC: Voltage V_{Batt} decreases monotonically as the battery is discharged, but the rate of decrease is not constant.
- The actual usable capacity of a battery cell depends on the discharge rate: At higher rates, the cell is less efficient at converting its chemically stored energy into available electrical energy.
- The "frequency" of the discharge current affects the amount of charge the battery can deliver: The battery does not react instantaneously to load changes, but it shows considerable inertia, caused by the large time constants typical of electro-chemical phenomena.
- Batteries operated at high discharge rate for a short period of time can recover available charge if the current load is temporarily reduced.

These effects can be modeled at the circuit level as shown in Figure 1.

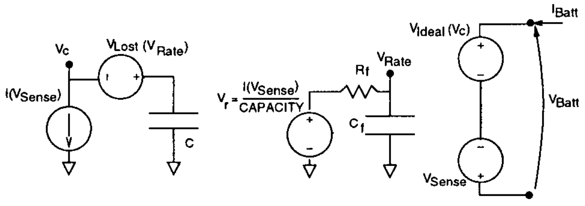

Figure 1: Continuous-Time Battery Model.

Dependency on SOC ($V_{Ideal}(V_C)$) is realized by storing several points of the curve in a look-up table (LUT) addressed by the value of the state of charge (V_C). The model is accurate up to a minimum *cut-off voltage*, after which the battery is considered fully discharged.

Dependency on discharge rate is modeled with a voltage source V_{Lost} in series with the charge storage capacitor. Voltage V_{Lost} reduces the apparent charge of the battery (which controls battery voltage (V_{Batt})). The value of V_{Lost} is a non-linear function of the discharge rate (which can be modeled by another LUT).

Dependency on the discharge frequency and the transient behavior of the battery, which includes the recovery effect, are modeled by averaging the instantaneous discharge rate used to control V_{Lost} through a low-pass filter (R_f, C_f). The low-pass filter models the relative insensitivity of batteries to high-frequency changes in the discharge current. Notice that V_{Sense} is a zero-valued voltage source added in series with the output voltage functions as the discharge-current (I_{Batt}) sensor. Additional effects such as temperature and internal resistance are also taken into account in order to increase the accuracy of the model.

According to [4, 5], this model fits measured data fairly well (within 15%). This accuracy is acceptable, since the actual capacity of any group of cells may vary as much as 20% between identical units, when we take into account manufacturing variances [4].

The discrete-time model of [3] can track the continuous-time one under various load conditions. In fact, the comparison of the estimates provided by the two models gives a 0.5% error for what concerns battery life-time analysis, and 0.7% when we consider battery output voltage results. The errors are mainly due to discretization. Furthermore, in the discrete-time model, the non-linear relationship between the two quantities is obtained by piece-wise linear approximation of values tabulated in an array. Conversely, in the continuous-time model the interpolation of the tabulated values is obtained by imposing the continuity of the first derivative.

3 Battery-Driven DPM Policies

DPM policies target the maximization of battery life-time by controlling the *mode of operation* of the system (or its components). Power-managed systems must be able to operate in different states which trade off performance for power consumption. We can distinguish between reduced-performance, low-power active states, and inactive, quiescent states. When the system is idle, the power management can force a transition to a quiescent state. The price to be paid for such a transition is latency and power. Generally, there is a delay and an energy cost for activating a system in quiescent state. Even if the system is not idle, in some cases, it can be forced in a state where it operates with lower performance and with reduced power.

3.1 Case Study: An MP3 Audio Player

To illustrate battery-driven DPM, we consider the system-level description of an MPEG 2-Layer 3 (MP3) digital audio player, whose block diagram, shown in Figure 2, is similar to a commercially-available product by Diamond [7]. System components can be power-managed through signals issued by a DPM unit in accordance with the selected DPM policy.

The MP3 player consists of a core processor (*ARM720T*) with 8 KB of cache and a DPM unit, 32 KB of static RAM (*SRAM*), an LCD controller (*LCD CTRL*), an MP3 codec (*CODEC*) and a memory controller (*MEM CTRL*); all these devices, together with some additional functionalities (e.g., interrupt controller), are contained in the EP7209

Ultra-Low-Power Audio Decoder System-on-Chip by Cirrus Logic [8]. External to the system there are a 32 MB flash memory (*FLASH*), the battery sub-system, a block (*DRIVER*) that emulates the inputs provided by the user, an LCD display and a head-set.

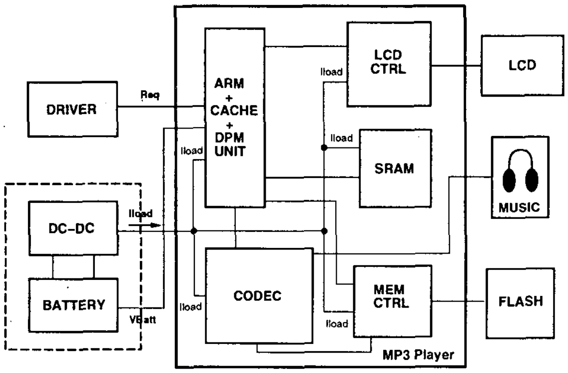

Figure 2: Block Diagram of the MP3 Player.

The system can operate in five different states:

- `Off`: The system is completely turned off and consumes no power.
- `Sleep`: The system is in sleep state and absorbs 33 mA.
- `Idle`: The system is idle and absorbs 38 mA.
- `RawMusic`: The system plays low-quality music and dissipates 46 mA.
- `FineMusic`: The system plays high-quality music and dissipates 57 mA.

When the system moves from one of the quiescent states (i.e., `Off`, `Sleep` and `Idle`) to one of the active states (i.e., `RawMusic` and `FineMusic`), it absorbs some additional current, as summarized in the following table:

	`RawMusic`	`FineMusic`
`Off`	23 mA	28 mA
`Sleep`	14 mA	17 mA
`Idle`	10 mA	11 mA

A typical usage of the system consists of an alternate, aperiodic sequence of active (playing music) and idle (silence) intervals. When no DPM policy is implemented, the system automatically enters the `Idle` state as soon as the `FineMusic` state is left (i.e., the song has terminated). From there, it can either go to the `Off` state, upon explicit request of the user, or go back to the `FineMusic` state, if a new song has to be played. Notice that states `Sleep` and `RawMusic` are never entered when the system runs in normal (i.e., non-power-managed) mode.

3.2 Open-Loop Time-Out Policy

We first consider a simple open-loop time-out policy. When the system stops playing, it enters immediately the `Idle` state; it waits there for a first time-out, T_1, then it transitions to the `Sleep` state. After a second time-out, T_2, if the system is still quiescent, it is transitioned to the `Off` state. Clearly, this policy aims at increasing battery lifetime by reducing the current absorbed by the system while it is not playing any music (states `Sleep` and `Off` are less current demanding than state `Idle`), but also by reducing the overhead due to transitions from states `Sleep` and `Off` to `FineMusic` (these states are not entered until time-outs have expired).

Notice that for the open-loop time-out policy discussed above, state `RawMusic` is not used. The duration of the time-out for each quiescent state is set to the ***break-even time*** (i.e., the minimum time to be spent in a quiescent state to amortize the energy spent in transitioning in and out of it). It was shown that this time-out choice is *2-competitive*, i.e., it can be outperformed by at most a factor of two by an oracle policy with complete knowledge of the future [6]. The time-out policy is *workload-driven* and it does not take into account battery characteristics.

3.3 Closed-Loop Policy

The simplest closed-loop policy is threshold-based. It aims at maximizing battery life-time by playing low-quality music when the battery is almost discharged. If the battery is fully charged, the system is kept in the `FineMusic` state. When the battery's output voltage falls below a threshold V_{Th}, the system is forced into the `RawMusic` state until the battery is fully discharged. The rationale for this policy is to provide graceful degradation of system performance as the battery discharges. Clearly, the choice of V_{Th} is critical for trading off music quality with battery life-time.

We have adopted the *quality factor* Q as quality metric. Q is defined as the ratio between the time the system is in the `FineMusic` state T_{Fine} and the total time of operation $T_{Fine} + T_{Raw}$. In symbols:

$$Q = \frac{T_{Fine}}{T_{Fine} + T_{Raw}}$$

The trade-off between life-time and music quality is captured by the product between (normalized) battery life-time and quality factor:

$$P = NLT \times Q$$

The optimal value V_{Th}^* that maximizes P depends on both system and battery characteristics. A complete exploration of the trade-off curve is provided in Section 4.

It is important to notice that the time-out and the voltage threshold policy are not mutually exclusive, and they should be applied together for best results. The hybrid policy exploits quiescent intervals in the workload, but it also trades off quality for battery life-time.

3.4 Policies for Multi-Battery Systems

Modern portable appliances, e.g., laptop computers, are able to accommodate two (or more) batteries in the same case. The batteries are used following a strict, sequential scheme: The second battery starts operating (i.e., supplying the current) only when the first battery is totally discharged.

A careful analysis of the time-domain model of a battery reveals that electro-chemical cells can recover some amount of deliverable charge if they are allowed to rest after a period of high-current discharge. This behavior can be fruitfully exploited in a two-battery system by adopting power management schemes where the two batteries alternate in providing current to the load. In this way, the battery temporarily disconnected from the load can recover, while the other one powers the system.

We study several open and closed-loop policies for two-battery power management. The baseline for the comparison is a two-battery system where batteries are discharged in sequence.

3.4.1 Open-Loop Switching Policy

A simple open-loop policy switches between one battery and the other with a fixed frequency f_{sw}. With this policy, that we call `Policy 1`, the life-time of the system depends on f_{sw}. For very low values of f_{sw}, each battery is drained for a long time with the full current load. The discharge behavior tends to the limiting case of $f_{sw} = 0$, in which the two batteries are discharged in sequence, one after the other. As f_{sw} increases, although the discharge behavior of the two batteries is less predictable, it is reasonable to expect a life-time increase thanks the recovery effect mentioned above.

This conjecture is confirmed by the experimental results reported in Section 4, which also include a detailed study for locating the most suitable value of f_{sw}. In principle, we would like to choose a value $f_{sw} \rightarrow \infty$, because the discharge behavior would tend to that of a single battery with double capacity. However, since the time constants of the batteries would reduce the amount of recovery the electro-chemical cells can benefit from, values of f_{sw} higher that f_{sw}^* (the critical switching frequency corresponding to the time constants of the batteries) would only marginally impact life-time extension. In addition, at a high f_{sw}, the behavior of the switching device that alternatively connects the batteries to the DC-DC converter may become a critical issue.

3.4.2 Closed-Loop Policies

A simple closed-loop policy can be obtained by setting a voltage threshold, as in the single-battery case. As soon as the output voltage of the battery system (i.e., the output voltage of the battery currently connected to the load) drops below threshold V_{Th}, the system is transitioned to the `RawMusic` state, until full discharge. The main shortcoming of this scheme, that we call `Policy 2`, is that it does not take into account the charge recovery of the batteries during the "rest period". Even if a battery output voltage drops below V_{Th} while the battery is fully loaded, it may raise back to a value higher than the threshold while the battery is unloaded. Hence, the simple threshold-based scheme may transition the system into `RawMusic` too early, thereby decreasing the quality of the played music.

This limitation of the basic closed-loop policy can be overcome if we adopt a slightly more complex switching scheme, called in the sequel `Policy 3`. More specifically, we propose a policy with three regions of operation. In the first region, the switching between the two batteries has constant frequency, and the state of operation is `FineMusic`. The second region is entered when the output voltage of one battery first reaches V_{Th}. The state of operation is still `FineMusic`, but switching between batteries is voltage-controlled. When the output voltage of the loaded battery reaches V_{Th}, it is disconnected from the load (to give it some recovery time). The second region is exited when the output voltage of the battery temporarily disconnected from the load does not increase beyond $V_{Th} - \Delta V$ during the recovery time. In the third region, the fixed frequency switching scheme is restored, and the system is transitioned into the `RawMusic` state until both batteries are fully discharged.

Needless to say, the two-battery policies are orthogonal with the workload-driven time-out policy. In other words, the time-out policy can be applied together with them, to synergically enhance life-time. In Section 4, the policies are implemented on the MP3 digital audio player, and their performance is compared.

4 Experiments

To collect all the experimental data, we have applied to the inputs of the system a work-load consisting of an input trace corresponding to a typical usage of the MP3 player over a time period of approximately one hour. Therefore, playing and silent intervals of different duration are interleaved in a non-correlated fashion, and are sometimes followed by shut-down commands issued directly by the user (which force the system to the `Off` state). In the sequel, we first present results for DPM of a single-battery system. Then, we discuss the case of a two-battery system.

4.1 Single-Battery System

The open-loop, time-out policy is the first solution we have tested out. It extends battery life-time from 2998 to 6643 seconds (that is, by approximately 121%). Application of the closed-loop policy first requires the identification of the threshold voltage, V_{th}, that discriminates between system operation in `FineMusic` and `RawMusic`. Figures 3, 4 and 5 show normalized battery life-time (NLT), quality factor (Q) and product, $P = NLT \times Q$ of these two quantities as functions of V_{th}.

As expected, NLT increases monotonically as V_{th} increases, while Q decreases, still monotonically but with a different shape and slope. Therefore, the product curve exhibits a maximum value for $V_{th} = V_{th}^* = 3.65V$. We used this value of V_{th} in the implementation of the battery-driven, closed-loop policy.

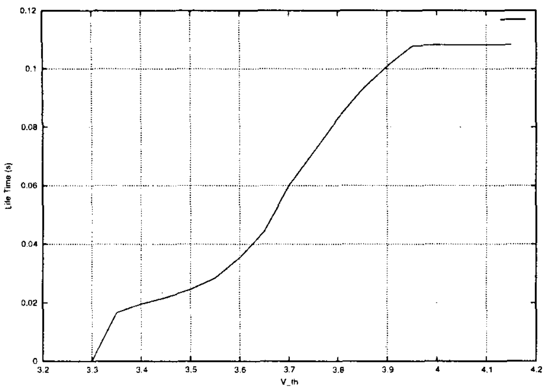

Figure 3: Normalized Battery Life-Time (NLT) vs. V_{th}.

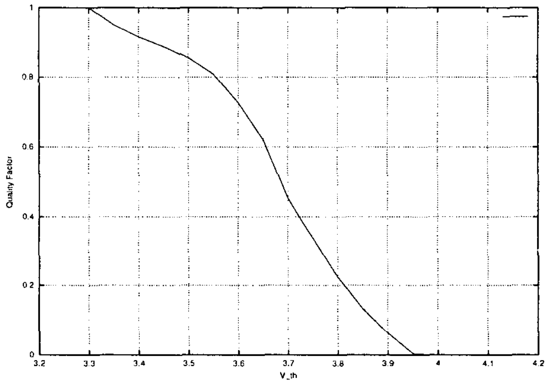

Figure 4: Quality Factor (Q) vs. V_{th}.

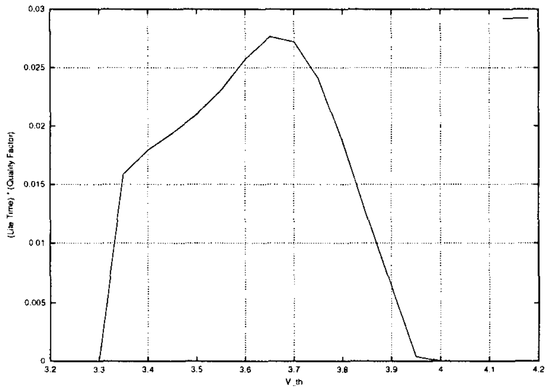

Figure 5: P vs. V_{th}.

When the policy is applied in isolation, i.e., with the time-out policy disabled, a life-time extension of 119% has been obtained (6612 seconds against 2998). As we have already noted in Section 3.3, the two policies are not mutually exclusive. Actually, they are very effective if they are combined together. Life-time extension has gone up to 6938, that is, 132% higher than the non-managed case. Values of the quality factor are acceptably high (0.688926 for the closed-loop policy alone and 0.622579 for the combined policy).

Clearly, a different choice of the threshold voltage would change both battery life-time and quality factor. Moving towards higher values of V_{th} would imply a longer duration of the battery at the cost of a reduced quality of the played music. The opposite would occur by decreasing the threshold voltage. This is demonstrated by the results of Table 1, in which life-time (in seconds) and Q are reported for different values of V_{th}, namely, $V_{th} = 3.65V$ (i.e., V^*_{th}), $V_{th} = 4.0225V$ and $V_{th} = 3.4275V$.

	Without Time-Out Policy		With Time-Out Policy	
V_{th}	LT	Q	LT	Q
$3.4275V$	6604	0.950062	6781	0.901163
$3.65V$	6612	0.688926	6938	0.622579
$4.0225V$	6873	0.000374	7362	0.000316

Table 1: Life-Time and Quality Factor for Different V_{th}.

We observe that, if the time-out policy is disabled, the difference in life-time extension between the case $V_{th} = 3.4275V$ and the case $V_{th} = 3.65V$ is almost negligible, while the quality factor is much higher. This indicates that the choice of the optimal value of V_{th} is not always advisable in practice.

4.2 Multi-Battery System

The first set of experiments performed on two-battery policies assesses the life-time extension that can be achieved by switching between two batteries with a fixed frequency f_{sw}. Battery life-time for different values of f_{sw} are shown in the semi-log diagram of Figure 6. The plot clearly indicates that the battery switching scheme results in sizable life-time extensions for a range of switching frequencies. When f_{sw} is very low, the two batteries are discharged in sequence, and life-time is minimum. This corresponds to the scheme currently adopted by commercially available appliances that contain more than one battery in the case. As f_{sw} increases, life-time increases as well, until a region of diminishing return is reached. Most of the life-time benefits are obtained by switching between batteries with $f_{sw} \approx 0.1Hz$. Observe that this frequency is very close to the battery time constant (i.e., the time required by the battery to respond to changes in current). Notice also that, in order to isolate the effect of f_{sw} on battery life-time, the curve of Figure 6 has been determined by loading the system with a constant current, instead of the usual trace of operation. In particular, the current load we applied corresponds to that absorbed by the system when running in the `FineMusic` state.

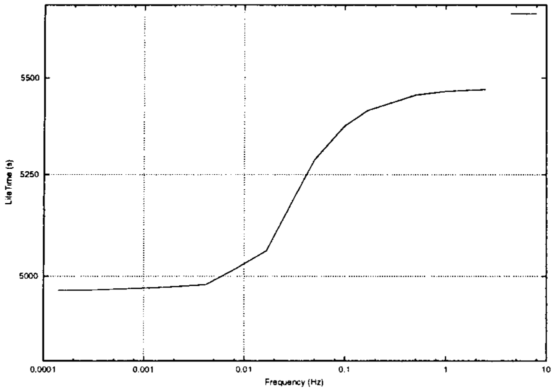

Figure 6: Battery Life-Time vs. f_{sw}.

A second set of experiments was performed to test the effectiveness of the policies presented in Section 3.4. The switching frequency was set to $f_{sw} = 0.1Hz$, and workload-driven power management was enabled to achieve maximum life-time extensions.

The results we have obtained are collected in Table 2. The open-loop policy (i.e., `Policy 1`) with $f_{sw} = 0.1Hz$, used as a baseline for comparison, has maximum quality, because the system never enters the `RawMusic` state. The first closed-loop policy (i.e., `Policy 2`), with a voltage threshold set to $V_{th} = 3.65V$, aggressively trades off quality for life-time extension, while the second closed-loop policy (i.e., `Policy 3`) slightly improves quality with a small penalty in life-time.

Policy	LT	ΔLT	Q	ΔQ
Policy 1	9448	–	1	–
Policy 2	10651	12.7%	0.6372162	-36.3%
Policy 3	10531	11.5%	0.6420397	-35.8%

Table 2: Life-Time and Quality Factor for Different Two-Battery Policies.

In summary, the results we have obtained demonstrate that two-battery switching policies effectively increase life-time, even when no quality loss can be tolerated. Furthermore, closed-loop policies can trade off quality losses for sizable life-time extensions.

5 Conclusions and Future Work

Battery-driven power management opens new opportunities for life-time extension in portable systems. In this paper we have proposed several open-loop and closed-loop policies that increase battery life-time by taking into account battery characteristics. The policies were validated on a test system (an MP3 digital audio player) with satisfactory results. Our experiments have also shown that battery-oriented power management can work synergically with traditional workload-driven DPM to achieve better life-time. Future work will focus on the study of automatic power optimization algorithms for battery operated systems, as well as on implementing battery-driven power management on real-life portable systems.

References

[1] L. Benini, A. Bogliolo, G. De Micheli, "Dynamic Power Management of Electronic Systems," *ICCAD-98: IEEE/ACM International Conference on Computer-Aided Design*, pp. 696-702, San Jose, CA, November 1998.

[2] L. Benini, A. Bogliolo, G. De Micheli, "System-Level Dynamic Power Management," *VOLTA-99: IEEE Alessandro Volta Memorial Workshop on Low-Power Design*, pp. 23-31, Como, Italy, March 1999.

[3] L. Benini, G. Castelli, A. Macii, E. Macii, M. Poncino, R. Scarsi, "A Discrete-Time Battery Model for High-Level Power Estimation," *DATE-00: IEEE 2000 Design Automation and Test in Europe*, pp. 35-39, Paris, France, March 2000.

[4] S. C. Hageman, "Simple PSpice Models Let You Simulate Common Battery Types," *EDN*, pp. 117-132, October 1993.

[5] S. Gold, "A PSPICE Macromodel for Lithium-Ion Batteries," *12th Annual Battery Conference on Applications and Advances*, pp. 215-222, January 1997.

[6] A. R. Karlin, M. S. Manasse, L. A. McGeoch, S. Owicki, "Competitive Randomized Algorithms for Non-Uniform Problems," *Algorithmica*, Vol. 11, No. 6, pp. 542-571, June 1994.

[7] Diamond Multimedia, *Rio PMP300*, http://www.diamondmm.de/eng/products/rio/rio300.htm.

[8] Cirrus Logic, *EP7209 Ultra-Low-Power Audio Decoder SoC*, http://www.cirrus.com/products/overviews/ep7209.html.