Optimizing the Power Delivery Network in a Smartphone Platform

Woojoo Lee, *Student Member, IEEE,* Yanzhi Wang, *Student Member, IEEE,* Donghwa Shin, *Member, IEEE,*
Naehyuck Chang, *Fellow, IEEE,* and Massoud Pedram, *Fellow, IEEE*

***Abstract*—Smartphones consume a significant amount of power. Indeed, they can hardly provide a full day of use between charging operations even with a 2000 mAh battery. While power minimization and dynamic power management techniques have been heavily explored to improve the power efficiency of modules (processors, memory, display, GPS, etc.) inside a smartphone platform, there is one critical factor that is often overlooked: the power conversion efficiency of the power delivery network (PDN). This paper focuses on dc–dc converters, which play a pivotal role in the PDN of the smartphone platform. Starting from detailed models of the dc–dc converter designs, two optimization methods are presented: 1) static switch sizing to maximize the efficiency of a dc–dc converter under statistical loading profiles and 2) dynamic switch modulation to achieve the high efficiency enhancement under dynamically varying load conditions. To verify the efficacy of the optimization methods in actual smartphone platforms, this paper also presents a characterization procedure for the PDN. The procedure is as follows: 1) group the modules in the smartphone platform together and use profiling to estimate their average and peak power consumption levels and 2) build an equivalent dc–dc converter model for the power delivery path from the battery source to each group of modules and use linear regression to estimate the conversion efficiency of the corresponding equivalent converter. Experimental results demonstrate that the static switch sizing can achieve 6% power conversion efficiency enhancement, which translates to 19% reduction in power loss general usage of the smartphone. The dynamic switch modulation accomplishes similar improvement at the same condition, while also achieving high efficiency enhancement in various load conditions.**

***Index Terms*—DC–DC power converter, low-power design, power delivery network (PDN), smartphone.**

I. Introduction

GROWING demand for increased smartphone functionality and the need to support all kinds of popular applications on the smartphone have been driving the trend toward including many high-performance modules (such as high-speed processors, fast wireless interface, large and high resolution display, sophisticated sensors) on the smartphone platform. The usability of smartphones has, however, been hampered by their low service time between successive charging operations. This is because the electrical energy storage density of modern batteries has been advancing at a relatively low pace compared to rate at which functional and performance improvements have been made to the smartphone platform and components. The latter, however, comes at the expense of increased power consumption in the smartphone platform.

Consequently, there has been a surge of interest in reducing power consumption of the smartphone platform. Some recent works have focused on developing power macromodels for the modules in the smartphone platforms [2]–[5]. Similarly, dynamic power management (DPM) techniques [6], [7] have been widely investigated and employed in various platforms, including smartphones.

While power modeling and DPM in the smartphone platforms have been heavily investigated, there is one critical factor that has often been overlooked, and that is the power conversion efficiency of the power delivery network (PDN) in smartphones. The PDN provides the battery power to all the modules. The conceptual diagram of the PDN in Fig. 1 shows that it consists of dc–dc converters. In reality, dc–dc converters in the PDN of a smartphone inevitably dissipate power, and power dissipations from all converters inside the platform can result in a considerable amount of power loss. Given that the overall PDN's power efficiency is the ratio of the power consumed by all the smartphone modules to the power drawn from the smartphone battery, Fig. 2 shows that the overall power efficiency of a real smartphone platform is around 60%–75%. Improving the power conversion efficiency can thus ensure appreciably longer battery life. This paper focuses on power conversion efficiency in the smartphone platform and introduces an optimization procedure for improving it.

Modern dc–dc converters exhibit high peak power conversion efficiency, but their efficiency can drop dramatically

Manuscript received March 28, 2013; revised July 7, 2013; accepted August 7, 2013. Date of current version December 16, 2013. This work was supported in part by grants from the Software and Hardware Foundations of the Division of Computer and Communication Foundations of the U.S. National Science Foundation, in part by the National Research Foundation of Korea (NRF) Grant funded by the Korean Government (MEST) under Grant 2012R1A6A3A03038938, and in part by the Center for Integrated Smart Sensors funded by the Ministry of Science, ICT and Future Planning as Global Frontier Project under Grant CISS-2012054193. The ICT at Seoul National University provided research facilities for this paper. This paper was presented in part at the 2012 International Symposium on Low Power Electronics and Design, Redondo Beach, CA, USA [1]. This paper was recommended by Associate Editor Y. Shin.

W. Lee, Y. Wang, and M. Pedram are with the Department of Electrical and Electronic Engineering, University of Southern California, Los Angeles, CA 90089 USA (e-mail: woojoole@usc.edu; yanzhiwa@usc.edu; pedram@usc.edu).

D. Shin is with the Dipartimento di Automatica e Informatica, Politecnico di Torino, Torino 10129, Italy (e-mail: donghwa.shin@polito.it).

N. Chang is with the Department of Electrical Engineering and Computer Science, Seoul National University, Seoul, Korea (e-mail: naehyuck@elpl.snu.ac.kr).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCAD.2013.2282287

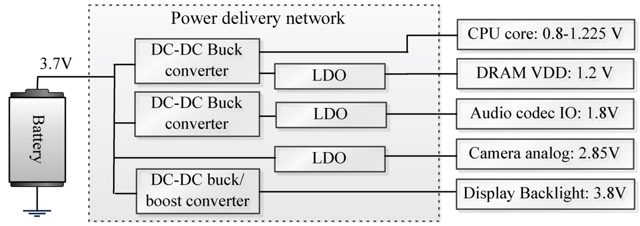


Fig. 1. Conceptual diagram of the PDN in a smartphone platform.

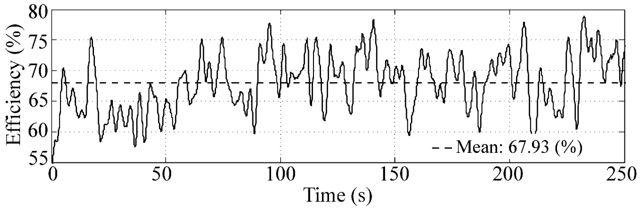


Fig. 2. Measure of traces of the power conversion efficiency of the PDN in the Qualcomm Snapdragon MDP MSM8660.

under adverse load conditions (i.e., out-of-range output current levels) [1], [8]. In other words, a state-of-the-art switching dc–dc converter can exhibit low conversion efficiency when there is a mismatch between the converter characteristics and its load. To tackle this drawback, a few approaches have been proposed. To design an optimal structure of the dc–dc converters, two methods have been introduced [9], [10]. A multiswitching scheme has been proposed to adaptively change the converter characteristics according to the load conditions [11], [12]. Component tuning in a dc–dc converter to ensure that the converter operates with high efficiency under the given load condition has been suggested in [1] and [13].

Starting with detailed models of the dc–dc converter designs, this paper presents two optimization methods to minimize the power loss due to the dc–dc converters according to the load conditions. First, we propose a static switch sizing (S3) method. The objective is to statically perform optimal sizing on the output stage drivers of the converter (i.e., the power MOSFET switches) at design time, according to statistical information about the load behavior. Next, we extend the multiswitching scheme to adaptively turn on/off the switches inside the dc–dc converter, depending on the required amount of load current. This method, called dynamic switch modulation (DSM), enables the dynamic control of the dc–dc converter so as to minimize its power loss under dynamically changing load conditions. This paper provides sophisticated control policies of the multiple switches as well as design optimization algorithms to find the number of switches and their optimum sizes.

To apply the proposed optimization methods to the actual smartphone platform, we perform the PDN characterization [1]. This paper proposes a characterization procedure, based on: 1) development of an equivalent dc–dc converter model; 2) module grouping; and 3) linear regression. The proposed equivalent dc–dc converter model can effectively model different types of converters and their cascade connections to represent a power delivery path from the battery cell to a collection of load devices. Each equivalent dc–dc converter model has its own conversion efficiency coefficients, and we perform characterization to identify these coefficients. The module grouping procedure enhances the accuracy of linear regression used for the conversion efficiency characterization.

This paper also provides extensive experimental results. We verify the accuracy of power conversion efficiency characterization with real measurement data. The results point out to the fact that power conversion efficiency of the target smartphone platform is quite low. Next, the load current profiles for each module in the smartphone platform are collected. Finally, we apply the two proposed optimization methods (i.e., S3 and DSM) to ensure that the converters operate at the most energy-efficient points. The experimental results demonstrate that the S3 achieves 6% overall efficiency enhancement, which translates to 19% power loss reduction for the general smartphone usage pattern. The results of DSM show that it can accomplish the efficiency enhancements as high as the S3. Furthermore, DSM can perform the efficiency enhancement for the whole load current range conditions.

The remainder of this paper is organized as follows. Section II provides some background on the dc–dc converter model. In Section III, the two optimization methods are presented. Section IV introduces the characterization procedures of the power conversion efficiency. Section V is dedicated to the experimental work, while Section VI concludes this paper.

II. DC−DC Converter Model

Typical dc–dc converters in the smartphone platforms can be classified into three types, inductive dc–dc converters, low-dropout linear regulator (LDO), and capacitive dc–dc converters, according to the circuit implementation and operation principles. The inductive dc–dc converters achieve very high power conversion efficiencies for wide range of their output loads. These types of converters can step up the output voltage so that it becomes higher than the input voltage (i.e., boost), or step down the output voltage so that it is lower than the input voltage (i.e., buck). On the other hand, the output voltage of an LDO can only be lower than its input voltage. In general, LDOs are good at low-noise output voltage, low area-overhead, and ease of integration. However, their limitation of low power conversion efficiencies makes them normally used to provide power for some noise-sensitive RF or analog modules in smartphones. The capacitive dc–dc converters have lower area overhead than the inductive dc–dc converters, and achieve better power conversion efficiency than LDOs. However, unlike the inductive dc–dc converters where the power conversion efficiencies depend only on parasitics of their components, the conversion efficiencies of the capacitive dc–dc converters are limited by their output resistance. Thus, it drops significantly as the conversion ratio moves away from the ideal ratio of a given topology and operating mode [14]. In this paper, we consider only the buck type inductive dc–dc converters and the LDOs. Using those dc–dc converters can appropriately provide the low-noise output voltages with high conversion efficiency to the various modules in the smartphone platforms.

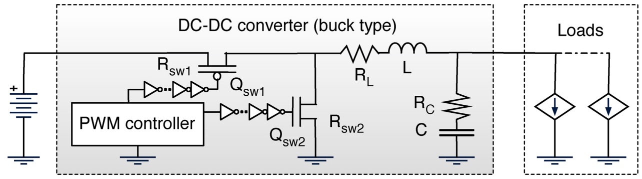


Fig. 3. Circuit diagram of a buck-type inductive dc–dc converter.

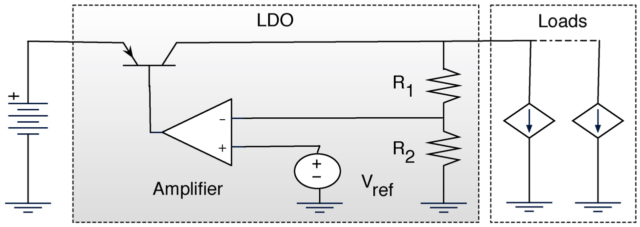


Fig. 4. Circuit diagram of a low-dropout linear regulator (LDO).

A. Inductive dc–dc Converter Model

The inductive dc–dc converter consists of an inductor, a capacitor, two MOSFET switches, and a pulse-width-modulation (PWM) controller. Fig. 3 shows the simplified schematics of the buck type inductive dc–dc converter (simply called dc–dc converter in the remainder of this paper). The PWM appropriately charges or discharges the output node to keep the output voltage of the converter at a desired target level. The high frequency switching noise is rejected by the L-C filter, whereas a small but important portion of the noise appears as output voltage ripples. Major power losses arise from the on-resistance of power switches and the parasitic resistance of passive elements in the design.

In Fig. 3, the pMOS switch is shown as *sw1*. Its ON-resistance and ON-state gate charge are denoted by R_{sw1} and Q_{sw1}, respectively. Similarly, the nMOS switch, shown as *sw2* in the figure, has an ON-resistance R_{sw2} and gate charge Q_{sw2}, accordingly. Parasitic series resistances of the inductor, L, and the capacitor, C, are denoted by R_L and R_C, respectively. Depending on the physical source of power consumption, the equation for the dc–dc converter power losses may be derived from the following three models: conduction loss, switching loss, and controller power consumption, denoted by $P_{\text{conduction}}$, $P_{\text{switching}}$, and $P_{\text{controller}}$, respectively [1], [8]. The power loss in the dc–dc converter, P_{loss}, is the sum of the three terms

$$\begin{aligned} P_{\text{inductive}} &= P_{\text{conduction}} + P_{\text{switching}} + P_{\text{controller}} & (1)\\ &= {I_{\text{out}}}^2(R_L + DR_{sw1} + (1-D)R_{sw2}) \\ &\quad +(\Delta I)^2(R_L + DR_{sw1} + (1-D)R_{sw2} + R_C)/12 \\ &\quad +V_{in} f_{sw}(Q_{sw1} + Q_{sw2}) + V_{\text{in}} I_{\text{controller}} & (2)\end{aligned}$$

where the first and second terms of (2) account for dc and ac conduction losses, respectively, the third and fourth terms of (2) are the switching loss and controller power consumption, respectively, I_{out} is the output current, V_{in} and V_{out} are the input and output voltages, D and $(1-D)$ are the PWM duty ratios of the pMOS and nMOS switches, respectively, f_{sw} is the switching frequency, $I_{\text{controller}}$ is the current used in the control logic section of the converter, and $\Delta I = (1-D)V_{\text{out}}/(L_f f_{sw})$ is the amplitude of the maximum current ripple at the inductor.

Finally, the conversion efficiency of a dc–dc buck converter, η, can be written as

$$\eta_{\text{inductive}} = \frac{P_{\text{out}}}{P_{\text{in}}} = \frac{V_{\text{out}} I_{\text{out}}}{V_{\text{out}} I_{\text{out}} + P_{\text{inductive}}} 100\ (\%). \quad (3)$$

From (2), the power losses due to the pMOS switch, P_{nMOS}, and nMOS switch, P_{nMOS}, may be expressed as

$$P_{\text{pMOS}} = C_{\text{ox}} W_p L_{\min} \frac{m}{m-1} V_{in}^2 f_{sw} + \frac{D I_{\text{out}}^2}{\mu_p C_{\text{ox}} \frac{W_p}{L_{\min}} (V_{\text{in}} - |V_{\text{pth}}|)} \quad (4)$$

$$P_{\text{nMOS}} = C_{\text{ox}} W_n L_{\min} \frac{m}{m-1} V_{in}^2 f_{sw} + \frac{(1-D) I_{\text{out}}^2}{\mu_n C_{ox} \frac{W_n}{L_{\min}} (V_{\text{in}} - V_{nth})}. \quad (5)$$

In (4) and (5), C_{ox} is the gate capacitance per unit area. W_p is the gate width of the pMOS power FET, and W_n is the gate width of the nMOS power FET. $L_{\min}$ is the minimum gate length of the given technology. μ_p is the hole mobility in the pMOS device, and μ_n is the electron mobility in the nMOS device. V_{pth} and V_{nth} are the threshold voltages of the pMOS and nMOS devices, respectively. m is the tapering factor for the (super buffer-like) gate driver of the power FETs. The output ripple of the converter, ΔV, is strictly limited by the normal operating conditions of the processor. Typically, ΔV must be less than 10% of the nominal output level. The PWM frequency, f_{sw}, and values of the passive components L and C significantly affect the magnitude of ΔV. Using the same notation as in the previous subsection, ΔV may be expressed as [15]

$$\Delta V = \frac{(V_{\text{out}} + V_{sw2} + V_L)(1 - \dfrac{V_{\text{out}} + V_{sw2} + V_L}{V_{in} - V_{sw1} + V_{sw2}})}{8LCf_{sw}^2} \quad (6)$$

where V_{sw1}, V_{sw2}, and V_L are the voltage-drops by $sw1$, $sw2$, and L, respectively.

According to (4), (5), and (6), the higher f_{sw} is, the smaller ΔV is, but the power dissipation $P_{switching}$ goes up. On the other hand, a smaller value of f_{sw} gives rise to a need for bigger L or C in order to meet the specified ΔV requirement.

B. LDO Power Loss Model

A typical LDO consists of an error amplifier, a pass transistor, and a feedback resistor network. The power loss of the LDO, denoted by P_{LDO}, is given by

$$P_{LDO} = I_{\text{out}}(V_{in} - V_{ref}\sigma) + I_q V_{in} \quad (7)$$

where V_{ref} is the reference voltage in the error amplifier, $\sigma = (R_1 + R_2)/R_2$ corresponds to the voltage divider's gain coefficient, and I_q denotes the quiescent current of the LDO. Unlike the switching converter in which the MOSFET switches dominate the total power loss, the pass transistor in

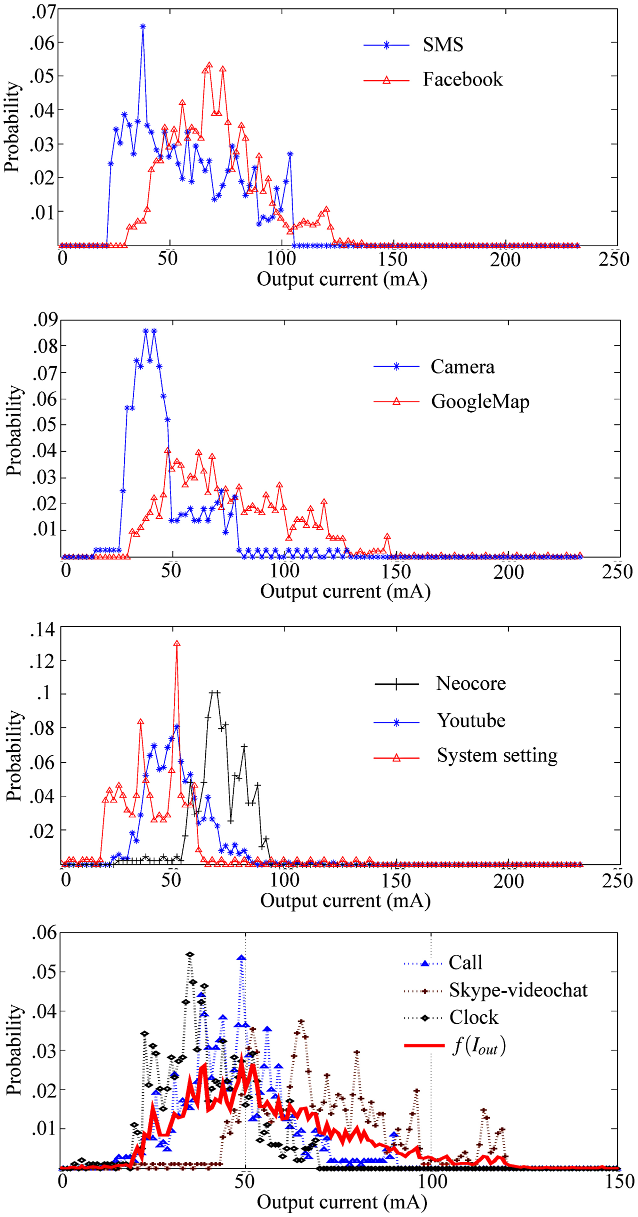


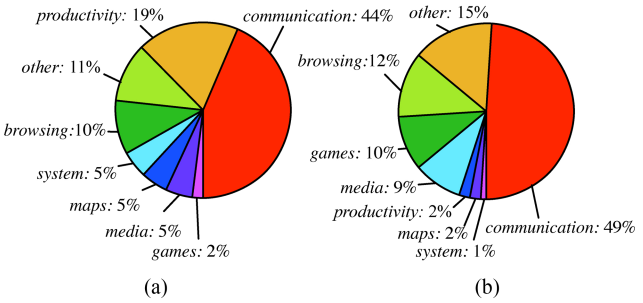
Fig. 5. Load current distributions of one core in MSM 8660 and a result of the derived $f(I_{\text{out}})$.

the LDO has a negligible impact on its total power loss [8]. Therefore, the power loss due to internal resistance of the pass transistor does not need to be explicitly accounted for in the model. Thus the conversion efficiency of the LDO, η_{LDO}, may be expressed as

$$\eta_{LDO} = \frac{V_{\text{out}} I_{\text{out}}}{V_{in} I_{in}} = \frac{\sigma V_{ref} I_{\text{out}}}{V_{in}(I_{\text{out}} + I_q)}. \tag{8}$$

III. DC–DC Converter Optimization

Optimizing dc–dc converters has the goal of reducing the power losses without incurring any performance degradation. This is because, unlike typical low-power design techniques that often exploit a tradeoff between performance, service



(a)

(b)

Fig. 6. Statistical data for the smartphone usages patterns, sourced from [19]. (a) Pattern I. (b) Pattern II.

quality, and power efficiency, the converter optimization technique does not shut off or slow down the overall system.

Enhancement of the overall efficiency of a dc–dc converter can greatly increase the overall system power efficiency [16], [17]. DC–DC converters show very high overall efficiency under desirable operating conditions. However, their efficiency can be low if they are operating outside the recommended range of input and output voltages and load currents [1], [8]. Therefore, ensuring that each dc–dc converter in the system is operating under the desirable operating conditions is an effective way of improving the system power efficiency. For example, [9] presents a dynamic programming-based approach to design the structure of the PDN in a system, while at the same time selecting the most suitable dc–dc converter or LDO for each node of the PDN. Reference [18] proposes the concept of parallel connections of high frequency dc–dc converters for distributed energy storage systems. In contrast, this paper starts with a fixed conversion tree structure, but performs MOSFET switch reconfiguration based on the load current demands and converter characteristics, so as to improve the overall power conversion efficiency in a smartphone platform.

A. Static Switch Sizing (S3)

Gate widths of the switches have a substantial impact on the efficiency of the dc–dc converter. From (4) and (5), P_{pMOS} and P_{nMOS} are convex functions of the change in gate width. The smaller gate width reduces the switching loss, but increases the conduction loss, and vice versa for the larger gate width. For a given I_{out}, the function to find the optimum pMOS gate width is thus obtained by solving $dP_{\text{pMOS}}/dW_p = 0$ [12], [19]

$$W_{p,opt}(I_{\text{out}}) = \frac{I_{\text{out}}}{C_{\text{ox}} V_{\text{in}}} \sqrt{\frac{D(m-1)}{\mu_p (V_{in} - |V_{pth}|) f_{sw} m}}. \tag{9}$$

The function to find the optimum nMOS gate width is derived in a similar manner, and its expression is as follows:

$$W_{n,opt}(I_{\text{out}}) = \frac{I_{\text{out}}}{C_{ox} V_{in}} \sqrt{\frac{(1-D)(m-1)}{\mu_n (V_{in} - V_{nth}) f_{sw} m}}. \tag{10}$$

It is important that the obtained optimum gate widths from (9) and (10) satisfy a design constraint whereby the resulting output ripple of the converter, ΔV, is less than its allowed limit. As described in (6), changing the switch sizes can affect ΔV. If the derived optimum switch sizes violate the ΔV constraint, we will increase L or C for the converter. Finally, the power loss of the converter in (2) is recalculated to ensure that the overall transistor sizing plus potential change to L and C reduce the net power loss. For reference, our experimental work in this paper shows that the worst case of ΔV increment from the default switch sizes to the optimum switch sizes is 14%. In other words, if ΔV for the default switch sizes is 5%, and then the resulting ΔV should be less than 5.7% (i.e., $5 + 5 \cdot 0.14$). We thus assume ΔV changes are enough small to satisfy the voltage ripple constraints. Detailed results of the ΔV increment are presented in Section V-D.

In (9) and (10), the optimum gate widths are derived for a fixed output current, I_{out}. However, I_{out} in the smartphone is different depending on its usage pattern. Therefore, the goal here is to find the optimum gate widths such that the high-conversion-efficiency operating conditions for the converter match with the current distribution that is produced by the actual usage profile of common smartphone applications. The optimization objective is thus to maximize the overall conversion efficiency of the smartphone based on its typical (expected) daily usage. Treating the total current used in the smartphone as a continuous random variable, we denote its probability density function by $f(I_{\text{out}})$. Because there are many mobile use cases generating various I_{out} distributions, finding a general case of $f(I_{\text{out}})$ is challenging. We propose a method utilizing the statistical data of mobile device usage patterns and measured data from running mobile applications as benchmarks as detailed next. First, we obtain a fine-grained classification of diverse mobile use cases. Next, we find mobile applications, representing each distinct class of use cases. We perform extensive measurement of output currents of the dc–dc converters in the smartphone platform when different applications are running. In addition, to derive the correct probability distribution of $f(I_{\text{out}})$, we acquire the average runtime of each class of use cases (applications) from the previous studies published in [20]–[22].

Fig. 5 shows example results of the derived $f(I_{\text{out}})$ distribution for a processor core in the Qualcomm's MDP. To derive $f(I_{\text{out}})$, we ran ten representative mobile applications (they are call, Facebook, Skype-videochat, clock, camera, Google-map, Neocore, SMS, system setting, and Youtube) on the MDP. Next, we classified the ten applications into seven classes presented in [21]: 1) communication (contains SMS, call, and Skype-videochat); 2) browsing (contains Web browsing); 3) media (contains camera and Youtube); 4) productivity (contains clock); 5) system (contains system setting); 6) games (contains Neocore); and 7) maps (includes Google Maps). We determined the average usage time of each class of applications based on the statistical data for the mobile device usage patterns [21]. As shown in Fig. 6, the reference introduced two representative smartphone use patterns (patterns I and II), each of which has its own proportions of the usage time for the aforesaid application classes.

From the derived $f(I_{\text{out}})$, (9) is modified to find the expected value of the optimum pMOS width from the S3 ($W_{p,S3}$)

$$W_{p,S3} = \frac{\sqrt{\int I_{\text{out}}^2 f(I_{\text{out}}) dI_{\text{out}}}}{C_{ox} V_{in}} \sqrt{\frac{D(m-1)}{\mu_p (V_{in} - |V_{pth}|) f_{sw} m}}. \quad (11)$$

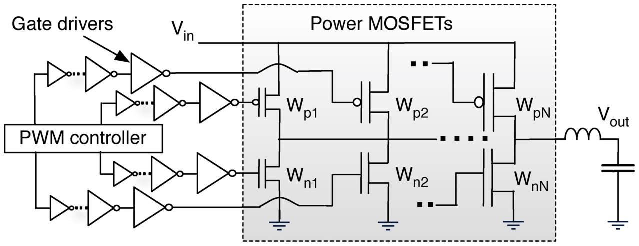


Fig. 7. Circuit diagram for dynamic switch modulation.

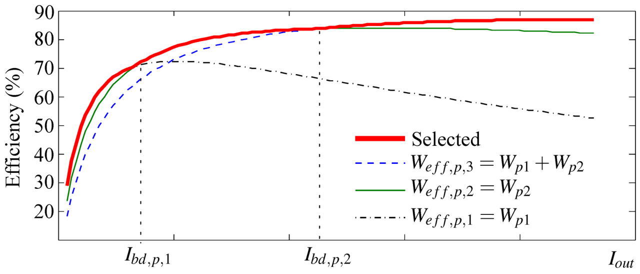


Fig. 8. Concept of DSM operation with two parallel-connected pMOS switches.

Similarly, the expected value of the optimum nMOS width from the S3 ($W_{n,S3}$) can be calculated as follows:

$$W_{n,S3} = \frac{\sqrt{\int I_{\text{out}}^2 f(I_{\text{out}}) dI_{\text{out}}}}{C_{ox} V_{in}} \sqrt{\frac{(D-1)(m-1)}{\mu_n (V_{in} - V_{nth}) f_{sw} m}}. \quad (12)$$

B. Dynamic Switch Modulation (DSM)

The S3 is only applicable when the load condition is given *a priori*. Any fixed sizing solution tends to result in suboptimal dc–dc conversion efficiency under dynamically changing load conditions, which may be very different from the one for which the static sizing solution was originally obtained. Furthermore, the higher the variance of the load current distribution is, the lower is the guarantee of optimality of the S3 solution. The optimum efficiency under dynamically changing load conditions can be obtained by adaptively turning on or off some of the multiple parallel-connected switches [11], [12]. However, the different gate voltages needed for each switch set in [11] require additional dc–dc converters, which tends to cause area/control overheads. Furthermore, the number of switches (which was fixed to three in [11] and [12]) and their sizes should be determined judiciously in order to achieve the maximum efficiency under given design specifications (i.e., for possible ranges of the load currents of various smartphone modules). Our proposed approach is an extension of the multiple switch scheme, which we call DSM. This task is to find the optimum number of parallel-connected output driver switches, their sizes, and on/off conditions under dynamically varying load conditions.

Fig. 7 shows a simple schematic drawing of the load-adaptive dc–dc converter. There are N pairs of switches connected in parallel. These switches are arranged such that the first switch has the minimum width (denoted by W_{p1} and W_{n1}), and the last switch has the maximum width (denoted

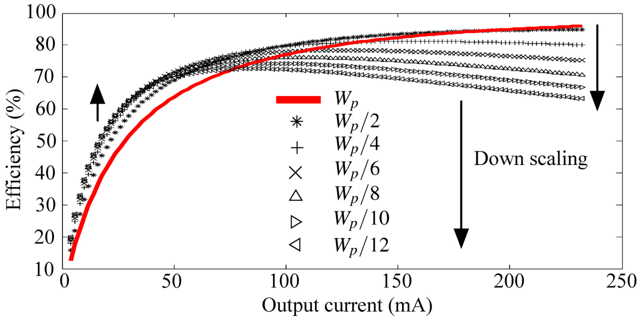


Fig. 9. Simulated power conversion efficiencies by changing the widths of the pMOS switch in Fig. 3.

by W_{pN} and W_{nN}). The maximum effective width (i.e., the sum of widths of all parallel-connected FETs of the same type) is large enough to support the maximum output current, $I_{\text{out,max}}$. For a smaller I_{out} value, some of the nMOS and pMOS switches are turned off. Depending on the I_{out} value, a different on/off combination of the switches can be used to achieve the maximum dc–dc conversion efficiency (which is equivalent to minimizing P_{pMOS} and P_{nMOS}).

We denote the effective width of the turned-on switch combination as $W_{eff,type,i}$, where *type* implies the switch type, i.e., p (pMOS) or n (nMOS), and i denotes the ith smallest effective width for the switch configuration (among all possible combinations of the same type of switch).

Fig. 8 is an example of the DSM on a dc–dc converter using two parallel-connected pMOS switches, which can independently be turned on or off at any time. The two pMOS switches give rise to three effective widths for the pMOS switch, $W_{eff,p,1}$, $W_{eff,p,2}$ and $W_{eff,p,3}$. Consequently, the output current range is divided into three operation ranges. The result of DSM in the figure, identified as a thick (red) line, shows that the maximum efficiency in each output current range is achieved by adaptively turning on the appropriate combination of two pMOS switches. It then follows that, for each output current range, the optimum switch combination must be found.

Note that the output current range can be divided into a larger number of bins by increasing the number of parallel-connected switches of the same type. A larger bin count greatly increases the flexibility to achieve high efficiency over a wider range of output current values. However, the increased area and power consumption due to higher complexity of the control circuitry is an important consideration in determining the optimal number of switches (N). To determine the optimum N and the size of each switch, we first investigate and determine the maximum and minimum effective widths of each type of switch. For the maximum effective width, we use the constraint that it should be large enough to drive $I_{out,max}$. Therefore, the maximum effective width of pMOS switch should satisfy the following constraint:

$$W_{eff,p,M} \geq \frac{I_{out,max} L_{min}}{\mu_p C_{ox}(V_{in} - |V_{pth}|)(V_{in} - V_{out,max} - R_L I_{out,max})} \tag{13}$$

where M is the number of all possible switch combinations (it is $2^N - 1$), $V_{\text{out,max}}$ is the maximum available output voltage of the dc–dc converter. $I_{\text{load,max}}$ can be obtained from measurements or looked up from a data sheet. We determine the maximum effective width of nMOS switches in a similar manner.

To determine the minimum size for the effective widths, we use our observation from the experimental work. Fig. 9 shows the result of simulating the dc–dc converter model in Fig. 3, for various widths of the pMOS switch. The model parameters are determined from the 45-nm BSIM4 predictive technology model for bulk CMOS [23], f_{sw} = 330 MHz, L = 6.8 nH, and C = 4 nF. According to the results, using switches smaller than a certain width region, yet it does not achieve high efficiency improvement even in the low current region. Therefore, the minimum effective widths should not be made too small.

Next, we consider the boundary conditions in the output current regions. We define the ith smallest boundary condition in the output current range as $I_{bd,type,i}$, where *type* is the switch type, while i is the ith smallest current value. Thus, $I_{bd,type,i}$ is the boundary condition between two consecutive switch combination regions, each of which has the corresponding optimum effective widths, $W_{eff,type,i}$ and $W_{eff,type,i+1}$. The example with the two pMOS switches in Fig. 8 shows that there are two boundary conditions, $I_{bd,p,1}$ and $I_{bd,p,2}$. From (4), the boundary condition for pMOS switches may be calculated as

$$I_{bd,p,i} = C_{ox} V_{in} \sqrt{\mu_p f_{sw} W_{eff,p,i} W_{eff,p,i+1} \frac{m}{D(m-1)}}. \tag{14}$$

The boundary condition for nMOS switches can be derived in the same way, and expressed as

$$I_{bd,n,i} = C_{ox} V_{in} \sqrt{\mu_p f_{sw} W_{eff,n,i} W_{eff,n,i+1} \frac{m}{(1-D)(m-1)}}. \tag{15}$$

Finally, we derive the objective functions for pMOS and nMOS sizing that minimize the expected power loss of pMOS (P_{pMOS}) and nMOS (P_{nMOS}) under the whole range of the possible output current values

$$\min \left(\sum_{i=1}^{M-1} \int_{I_{bd,p,i}}^{I_{bd,p,i+1}} \left(\alpha W_{eff,p,i} + \frac{D I_{\text{out}}^2}{\beta \ W_{eff,p,i}} \right) f(I_{\text{out}}) dI_{\text{out}} \right) \tag{16}$$

$$\min \left(\sum_{i=1}^{M-1} \int_{I_{bd,n,i}}^{I_{bd,n,i+1}} \left(\alpha W_{eff,n,i} + \frac{(1-D) I_{\text{out}}^2}{\gamma \ W_{eff,n,i}} \right) f(I_{\text{out}}) dI_{\text{out}} \right) \tag{17}$$

where $\alpha = C_{ox} L_{min} f_{sw} V_{in}^2 m/(m-1)$, $\beta = \mu_p C_{ox}(V_{in} - |V_{pth}|)/L_{min}$ and $\gamma = \mu_n C_{ox}(V_{in} - V_{nth})/L_{min}$. $I_{bd,p,M}$ and $I_{bd,n,M}$ are equal to $I_{out,max}$, whereas $I_{bd,p,1}$ and $I_{bd,n,1}$ equal the minimum output current. $f(I_{\text{out}})$ is the load current distribution.

Solving (16) and (17) is not straightforward. This is primarily because, as we also stated before, the number of possible combinations (M) increases exponentially as the number of switches (N) grows. In addition, we also have to abide by other design considerations, e.g., limitations on the control complexity and area overhead. Therefore, N should be carefully selected, i.e., it must be small enough so as not to significantly increase the control and area overheads, but large enough to enable the DSM in response to varying load conditions. Even if

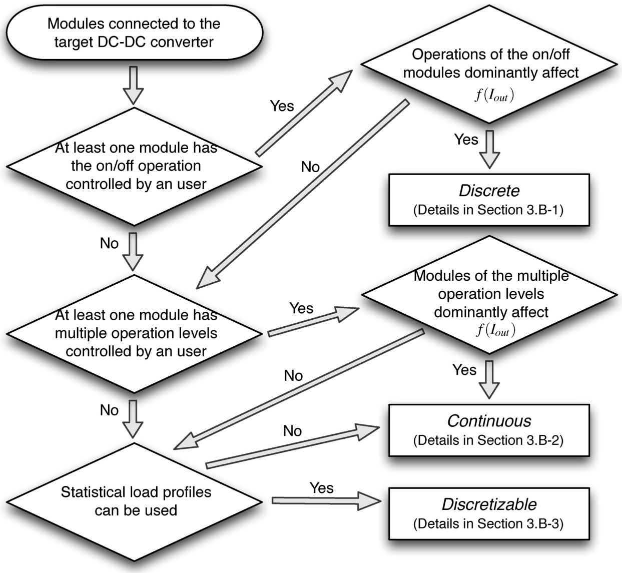

Modules connected to the target DC-DC converter
At least one module has the on/off operation controlled by an user
Yes
Operations of the on/off modules dominantly affect $f(I_{out})$
Yes
Discrete (Details in Section 3.B-1)
No
No
At least one module has multiple operation levels controlled by an user
Yes
Modules of the multiple operation levels dominantly affect $f(I_{out})$
Yes
Continuous (Details in Section 3.B-2)
No
No
No
Statistical load profiles can be used
Yes
Discretizable (Details in Section 3.B-3)

Fig. 10. Flowchart to classify $f(I_{\text{out}})$ into three different cases.

N is limited to a small number, assuming that $f(I_{\text{out}})$ follows a uniform distribution may not guarantee the optimality of the solution. This is because actual load conditions can be discretely distributed (i.e., those modules which have ON/OFF operation controlled by user activities, such as camera, SD card and so on. We thus classify $f(I_{\text{out}})$ into three cases discrete, continuous, and discretizable, as described in Fig. 10. We present the heuristic solution of the switch selection and sizing problem for each case in the following subsections.

1) *Discrete* $f(I_{\text{out}})$*:* We define the state of $f(I_{\text{out}})$ as discrete when $f(I_{\text{out}})$ has (discretely) dominant load current values. For example, if a dc–dc converter powers up some modules including modules that can be controllably turned on/off, it may have several discrete load current values in its $f(I_{\text{out}})$. If the discrete values are dominant in the distribution, the problem here aims to select and size the switches so that the effective widths of the switches match to the widths corresponding to the discrete current values, calculated by (9) and (10). According to the switch type, the calculated widths are included to a set G_p (for pMOS) or G_n (for nMOS). We then define cover so that a set S covers a width w means there is an effective width configured by elements in S to match to the value, $w \pm \Delta$. Δ should be small enough. If the given design specification has enough switches (N) so that the effective widths can easily cover all the required widths in G_p and G_n; then, the problem can be solved straightforwardly. However, N is likely quite small in a common design specification. With the given N, the problem is then to find a minimum set of each switch types (T_p and T_n, where $|T_p|$, $|T_n| \leq N$) that can cover the maximum number of the widths in G_p and G_n. Finally, we present an algorithm to solve the problem. A function, *coverage*, in Algorithm III-B1 is a simple dynamic programming that determines whether the current set of switches (S) can cover the required width (w). Performing *OptP_widths* in Algorithm III-B1 returns the set, T_p, that cover the maximum number of elements in G_p. The optimum set for the nMOS switches can be obtained in a similar manner.

Algorithm 1 To find a minimum set of the optimum widths of PMOS switches (T_p) under the given number of switches (N) and the discretized I_{out}

```
Initialization
define $I_{out,i}$, $N$, $\Delta$                    ▷ $I_{out,i}$ is $i^{th}$ discrete current value
$W_i = W_{p,opt}(I_{out,i})$ and $G_p = \{W_1, W_2, ..., W_K\}$            ▷ from (9)
$max \leftarrow 0$      ▷ $max$ will be updated to the maximum number of
    elements in $G_p$, covered by the set $T_p$ from $OptP\_widths$

function $coverage(w, S)$
    if $|w| \leq \Delta$ then return 1
    for each $s \in S$ do
        if $coverage(w - s, S \bigcap \{s\}^c) = 1$ then return 1
    return 0

function $OptP\_widths(n, m, S)$                         ▷ : main function
    for $n \leq i \leq N$ do      ▷ $i$ is the number of switches in the set
        for $m \leq j \leq K$ do                 ▷ to add $W_j$ into the set S
            $S \leftarrow S \bigcup \{W_j\}$, $c \leftarrow 0$
            for $1 \leq k \leq K$ do                            ▷ to check $W_k$
                if $coverage(W_k, S) = 1$ then $c \leftarrow c + 1$
                else if $|S| < i$ and $j \leq k$ then
                    $OptP\_widths$(i,k,S)
                    $c \leftarrow c + 1$
            if $\sum_{s \in S} s \geq W_{eff,p,M}$ and $c \geq max$ then
                $max \leftarrow c$, $T_p \leftarrow S$           ▷ to update $max$ and $T_p$
        if $max = K$ then break
    return $T_p$
```

2) *Continuous* $f(I_{\text{out}})$*:* Some dc–dc converters power up modules that have more than two operation levels as set by the user preferences. The brightness level of the display module and the volume level of the speaker module can be representative examples. If the load current of each module's operation level is known, then $f(I_{\text{out}})$ may belong to the discrete case. However, our experience with the Qualcomm MDP shows that the load current conditions of the various operation levels typically overlap. We thus cannot find discrete breakpoints in $f(I_{\text{out}})$. Furthermore, the user preference is random so that all the load current conditions have the same probability to be chosen. Finally, we define this case as continuous, and treat $f(I_{\text{out}})$ as an uniform distribution. In this case, finding a set of the effective widths (G_p or G_n) can be formulated as a simple arithmetic progression problem to find M number of effective widths with the given minimum and maximum effective widths. Next, Algorithm III-B1 is applied to the resultant set of effective widths so that we can find the switches to cover the maximum number of the effective widths.

3) *Discretizable* $f(I_{\text{out}})$*:* There are some dc–dc converters supplying power to the modules that cannot be controlled by the user. In this case, we propose to use the statistical load profiles of the dc–dc converter. Therefore, not only can the dc–dc converter deal with the dynamically varying load conditions, but also it has more possibility of being tuned for actual load conditions, compiled from the typical smartphone use patterns. The way to obtain the statistical load profiles is aforementioned in Section III-A.

We propose an approach to adapt K-means clustering to extract some discrete values from the load current values (I_{out}). The measured data of I_{out} is initially modified to I'_{out} that ith value of I_{out} is $\lambda f(I_{\text{out}})$ times duplicated in I'_{out}. λ is a factor

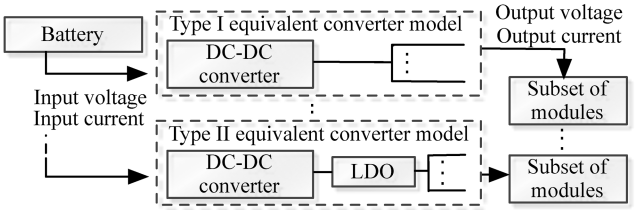


Fig. 11. Types I and II equivalent converter models.

to adjust the weight of $f(I_{\text{out}})$. Then, I'_{out} is divided into K parts evenly, and the initial means of all parts are calculated. For the update procedure in the K-means clustering, the new means, set to be the centroids of the parts, are calculated until the result of the means converges. Finally, the set of the resultant means for each type of switches become G_p and G_n, respectively. Then, they are applied to Algorithm III-B1 to find the minimum switch set that covers the maximum number of elements in G_p and G_n.

IV. Power Delivery Network Characterization

Prior to verifying the efficacy of the proposed dc–dc converter optimization methods in an actual smartphone platform, the power conversion efficiency of the PDN in the target platform should be characterized. However, the characterization is not a trivial task unless the PDN structure and converter specifications, and all the node voltages and branch currents of the PDN are available. Such a white-box approach is generally not possible for commercial smartphone platforms.

In this paper, we attempt a gray-box approach by introducing an equivalent converter concept. Modules in the platform are powered through the PDN, composed of a set of converters, as shown in Fig. 11. The converter set can be an empty set (direct connection), single dc–dc converter, a cascade connection of a dc–dc converter and an LDO, (rarely) a cascade connection of multiple dc–dc converters, etc. The equivalent converter models the set of converters on the path from the battery source to each (set of) module. In other words, the proposed equivalent converter abstraction treats the set of converters as a single equivalent converter. The abstraction enables a gray-box approach by which one can group modules in a smartphone platform by their required supply voltage levels, which can be obtained from datasheets. Power conversion efficiency improvement by adapting the proposed dc–dc converter optimization methods can effectively be performed once we identify the power conversion efficiency of the PDN in the smartphone platform.

A. Equivalent Converter Model

We classify the equivalent converter models to present either a single dc–dc converter, or a cascaded connection of a dc–dc converter and an LDO, called type I and type II equivalent converters, respectively. We assume that the battery output current flows through a voltage regulator in order to produce a constant voltage throughout full discharge cycle of the battery. Without loss of generality, types I and II equivalent converter models can represent most power conversion tree structures in the PDN [9], [17], [24]. Most digital logic components can be powered by a single dc–dc converter from the battery to the module—this gives rise to type I converter model. A cascade connection of two or more dc–dc converters is rare, because increasing the number of cascaded dc–dc converters generally increases the cost and area overhead with little (or no) benefit in terms of the conversion efficiency. LDOs are often an indispensable component to provide low-ripple output voltage for switching noise-sensitive RF and analog modules. It is uncommon to use a single LDO from the battery to a load device due to the required large dropout voltage and hence loss of LDO power efficiency. Instead, it turns out to be more energy-efficient to first convert the battery voltage using a dc–dc converter to an internal voltage slightly higher than the device voltage, and subsequently, use an LDO for the final power conversion.

According to (2) and (7), the power loss of the equivalent converter may be expressed as

$$P_{eqv} = A(\delta\ I_q + \sum_{i=1}^{N} I_{mod,i})^2 + \delta\zeta \sum_{i=1}^{N} I_{mod,i} + (B + \delta\nu I_q) \quad (18)$$

where N is the number of modules connected to the equivalent converter, $I_{mod,i}$ is the input current of the ith module, parameter A for the dc–dc converter is given by $A = R_L + DR_{sw1} + (1-D)R_{sw2}$, B is the sum of the second, third, and last terms of (2), $\delta = 0$ for type I, and $\delta = 1$ for type II equivalent converter, ν is the input voltages of the LDO, and $\zeta = (\nu - V_{ref}k)$. We can further simplify (18) by defining the output current of the equivalent converter, $I_{eqv\_out} = \sum_{i=1}^{N} I_{mod,i}$, and thus, the power loss for both types of equivalent converter models can be expressed as

$$P_{eqv} = aI_{eqv\_out}{}^2 + bI_{eqv\_out} + c \quad (19)$$

where the coefficients a, b, and c are derived from (18), and are largely dependent on the converter design specification such as the power MOSFET gate width, inductor IR loss, controller loss [8]. Calculating those coefficients is the key step of the power conversion efficiency characterization.

B. Module Grouping and Regression Analysis

Measurement (or estimation) of the output current of all the equivalent converters enables us to estimate the unknown coefficients of the equivalent converter model. The input and output voltage levels of each equivalent converter can be obtained from the device datasheets. For example, the Qualcomm MDP MSM8660 [25] incorporates embedded power sensors that monitor and report current values of each module in the platform with fine granularity. When the target platform does not provide embedded current sensors, we can estimate the module current values by activity profiling [2]–[5].

Profiling various applications, which result in diverse usage patterns of the system modules, provides sufficient information and data to perform regression analysis and estimate the unknown coefficients. Linear regression analysis is a widely used method in system identification, requiring: 1) a well designed model and 2) sufficient experimental data to extract the best-fit model coefficients. In reality, however, independent control of each module is a challenging task due to the lack

TABLE I

Grouping Results for Qualcomm MDP MSM8660

Group	Modules	Voltage
1 and 2	Group 1: CPU core0 and Group 2: CPU core1	0.8 - 1.225 V
3	Internal Memory, Audio DSP, and Digital core (includes GPU and modems)	1.1 V
4	Audio codec Vdd, LPDDR2, ISM, DRAM, and Camera-digital	1.2 V
5	Audio codec IO, IO PAD3, Display IO, DRAM Vdd1, Camera IO, PLL, and eMMC host interface	1.8 V
6	Camera analog, Haptic, SD card, Touch screen, eMMC (Flash), IO PAD2, SD card, and Ambient light sensor	2.85 V
7	Display memory and Display backlight	3.8 V

TABLE II

Extracted Coefficients for Each Group

k	a_k	b_k	c_k	k	a_k	b_k	c_k
1,2	0.4427	0.0025	0.0170	5	0.1971	0.5232	0.0128
3	0.4079	0.1742	0.0675	6	0.1814	0.2928	0.0320
4	0.1152	0.1757	0.0077	7	0.4091	0.3871	0.0289

of direct control knobs. For example, if we run an application that activates a camera module, currents flowing into the CPU, GPU, memory, and other associated components also ramp up and down correspondingly. We must thus apply linear regression analysis to the whole system (including all smartphone modules) simultaneously, while trying to vary the activity level of each module by running different applications. However, this method may not produce sufficient data to cover the whole range of activities for all smartphone modules, especially when the number of modules is large (e.g., the Qualcomm MDP has 27 embedded modules.) This is a potential source of inaccuracy for regression analysis due to the weak training set issue.

We tackle the problem by performing a module grouping in order to reduce the number of unknown coefficients that must be determined during the characterization process. This grouping procedure reduces the burden in terms of generating sufficient data to perform the linear regression analysis. The idea is that system modules that require the same operating voltage level can be combined into one group, and each group of modules is connected to the battery source via a single equivalent converter, as illustrated in Fig. 11. This method matches well with low power design practices that try to minimize the number of converters, due to their cost and internal power losses.

Given that the number of different voltage levels required by various modules in a smartphone platform is typically less than 10 [17], [24], the grouping procedure significantly reduces the number of parameters to be determined in linear regression. For example, the classification result of the Qualcomm MDP in Table 1 shows that the platform requires only seven groups although the module count is 27.

Finally, the total power loss of the smartphone, P_{loss}, is given by

$$P_{loss} = \sum_{k=1}^{G} P_{eqv,k} = \sum_{k=1}^{G} (a_k I_{eqv\_out,k}{}^2 + b_k I_{eqv\_out,k} + c_k) \quad (20)$$

where G is the number of groups, $P_{eqv,k}$ is the power loss of the kth equivalent converter corresponding to the kth group of modules, $I_{eqv\_out,k}$ denotes the output current of the equivalent converter, which can be measured using embedded sensors in the Qualcomm MDP, a_k, b_k, and c_k are the coefficients of the equivalent converter model (to be determined by linear regression). We treat the battery voltage presented to the power conversion tree as being (nearly) constant, which is valid considering the function of the regulator between the battery cell/pack and the equivalent converter. Therefore, we may assume that a_k, b_k, and c_k are constant values.

V. Experimental Work

A. Experimental Setup

Qualcomm MDP MSM8660 is used as an actual smartphone platform, which is equipped with Google Android OS 2.3 on top of Snapdragon 1.5 GHz asynchronous dual-core CPU, a 3D-supporting GPU, 3.61 in WVGA multitouch screen, 1-GB internal RAM, 16-GB on-board flash, WiFi, Bluetooth, a GPS, dual-side cameras, etc. We perform power measurement of each module using the application profiling tool called Trepn. Use of Trepn ensures higher accuracy of the measurements. Note, however, that our proposed method is independent of the measurement tools, e.g., we may use activity profiling for power measurement provided by Google or based on techniques presented in the literature [2]–[5]. The collected data from MDP8660 is next processed by MATLAB for the characterization, as well as the optimization procedures.

B. Coefficient Identification

As shown in Table I, the Qualcomm MDP modules can be classified into seven groups based on their operating voltage levels. Some modules such as the CPU cores in the MDP use dynamic voltage and frequency scaling techniques that require a range of variable supply voltage levels. Consequently, we keep each CPU core in a separate group but treat the equivalent converters of these groups identical to each other. Group 7 is associated with display, and therefore, the backlight brightness level mostly determines the current demand in this group. Group 7 coefficients are easy to identify because we can independently control the brightness of the display. In other words, we first perform the linear regression to identify coefficients of the equivalent converter model of Group 7, separately from the other groups.

For the remaining six groups, we profile various applications and collect sufficient data for the regression analysis as explained earlier. It is difficult to identify every c_k coefficient of the kth equivalent converters directly from the linear regression process. Rather, we only extract c_{ext} that corresponds to the sum of all the constant terms in (20), i.e., $c_{ext} = \sum_{k=1}^{G} c_k$. We find an approximate value for each c_k as $c_k = c_{ext}(P_{group,k}/P_{group,total})$, where $P_{group,k}$ denotes the power consumption of Group k, and $P_{group,total}$ is the total power consumption of all the groups. The $P_{group,k}$ and $P_{group,total}$ values are available from the embedded sensors in the MDP.

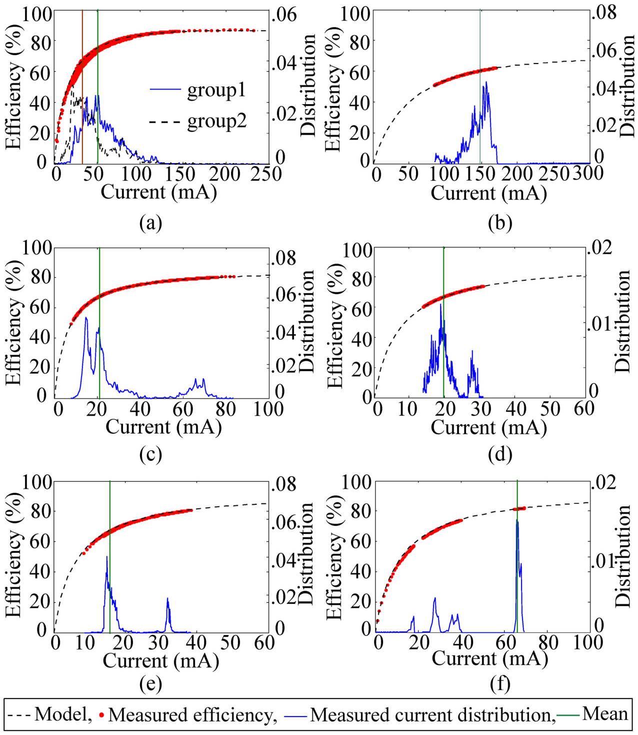


Fig. 12. Conversion efficiencies for all groups. (a) Groups 1 and 2. (b) Group 3. (c) Group 4. (d) Group 5. (e) Group 6. (f) Group 7.

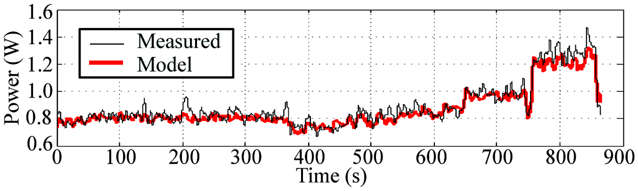


Fig. 13. Part of traces of total power consumption: measured data and modeled data.

The extracted coefficients of the seven equivalent converters are reported in Table II. The power conversion efficiency of Group k, derived from $(P_{group,k}/(P_{group,k}+P_{eqv,k}))$, is shown in Fig. 12. We verify the characterization results of each equivalent converters. Fig. 13 shows the comparison of the system power consumption trace between the real measurement as reported by a built-in battery sensor and the estimation as obtained by our extracted equivalent converter coefficients. The trace includes ten mobile applications, as stated in Section III-A. We measure the error as a signal-to-noise ratio, and the resulting average error is 0.075. The standard deviation of the error is 0.059. The worst case average error is 0.128 and is seen for Neocore (there is a rare but important synchronization problem with the built-in sensor causes extreme worst error in this case). We also run four completely new mobile benchmarks (they are different from the one used for the regression analysis): Antutu [26], [27], Quadrant [28], and GLBenchmark [29]. These benchmarks are designed to test the performance of various modules in the smartphone platform. In particular, Vellamo includes HTML5 and METAL chapter to evaluate the mobile web browser performance and the mobile processors, respectively. GLBenchmark and Antutu include a 3D testing for GPU. Quadrant performs CPU, Memory, I/O testings. Therefore, we believe these four new benchmarks are sufficient to evaluate our regression analysis. The resulting average error and standard deviation are 0.047, 0.046 for Antutu, 0.062, 0.040 for Vellamo:Metal, 0.092, 0.052 for Vellamo:Html5, 0.064, 0.045 for Quadrant, and 0.065, 0.058 for GLBenchmark. We have thus confirmed that the results of the power conversion efficiency characterization process is accurate enough for the subsequent optimization process.

TABLE III
W_{def} OF EQUIVALENT CONVERTER MODELS

Group	1, 2	3	4	5	6	7
W_{def}	1.2401	1.1033	1.3109	1.4033	1.4102	0.7368

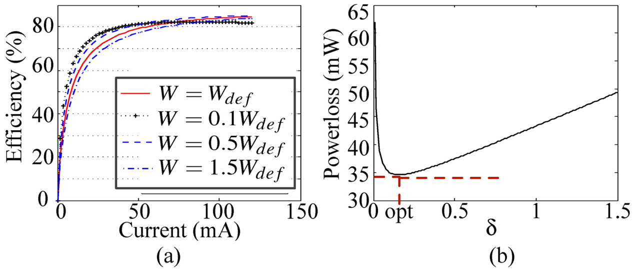


Fig. 14. Relation between the power conversion efficiency and W: Group 7. (a) Efficiency: Group 7. (b) Power loss: Group 7.

C. Default Widths Extraction

Given that the pMOS switch typically has smaller current per width than the nMOS switch, the pMOS switch is much larger than nMOS switch in the dc–dc converters [12]. We thus focus on scaling the width of the pMOS switch, and the nMOS switch is sized to have the same resistance of the pMOS switch (i.e., the widths of both switches are in turn linearly proportional to each other). From (4) and (5), the dc–dc converter power loss model may be generally expressed as a function of W

$$P_{converter} = \left(\frac{r_1}{W} + r_2\right) I_{\text{out}}^2 + r_3 W + r_4 \tag{21}$$

where W is linearly proportional to both width of pMOS and nMOS switches, r_1, r_2, r_3, and r_4 are constants.

Given that the two MOSFET switches in a dc–dc converter dominate the power loss of the equivalent converter, and I_q is small, we rewrite (19) as

$$P_{eqv,k} = \left(\frac{r_{1,k}}{W_k} + r_{2,k}\right) I_{eqv\_out,k}^2 + bI_{eqv\_out,k} + r_{3,k} W_k + r_{4,k} \tag{22}$$

where $P_{eqv,k}$ and $I_{eqv\_out,k}$ are the power loss and the output current of the kth equivalent converter, respectively, corresponding to the kth group of modules, W_k, $r_{1,k}$, $r_{2,k}$, $r_{3,k}$ and $r_{4,k}$ are the coefficients of the equivalent converter model that have been determined based on linear regression. In the linear regression procedure, we carefully set the initial condition not be trapped in a local minimum. Then, the resultant coefficient W_k is the default value of W (W_{def}) of the kth equivalent converter, which is shown in Table III.

D. Simulation Results: Static Switch Sizing

Fig. 14(a) shows an example in which W changes the efficiency graph of Group 7. Fig. 14(b) shows that the power loss plots have a convex functional form in terms of W. From (11) and (22), the optimal W of the kth group is calculated by

$$W_{opt,k} = \sqrt{\int\int_{I_{eqv\_out,k}} I_{\text{out}}^2 f_k(I_{\text{out}}) dI_{\text{out}}} \sqrt{\frac{r_{1,k}}{r_{3,k}}} \qquad (23)$$

where $f_k(I_{\text{out}})$ is the kth load current distribution.

In order to derive $f_k(I_{\text{out}})$, we use the collected loading profiles. As introduced in Section III-A, we run the ten representative mobile applications, and the loading profiles of all the modules in kth groups are measured for each application. All the applications except clock and system setting are run under the same setup where WiFi is turned on and the backlight level of the display is the highest. Clock is measured under the median level of the backlight and WiFi on, whereas system setting is measured under the lowest backlight and WiFi off. For the case of call, we consider auto turn-off screen during the call. We derive two types of load current distributions, according to the two representative smartphone usage patterns, patterns I and II introduced in Fig. 6. Fig. 12 shows the resulted $f_k(I_{\text{out}})$ from pattern I.

From (22), the expected power loss of an equivalent converter can be generally expressed as

$$E[P_{eqv}] = \left(\frac{r_1}{W} + r_2\right) \int I_{\text{out}}^2 f(I_{\text{out}}) dI_{\text{out}} + b \int I_{\text{out}} f(I_{\text{out}}) dI_{\text{out}} + r_3 W + r_4. \qquad (24)$$

We denote the efficiency and power loss for different setup as η_{setup} and P_{setup}, where *setup* can be *def* or *opt*. *def* implies the default setup, whereas *opt* implies the optimal setup of the dc–dc converter. Then, η_{setup} can be calculated by $P_{group}/(P_{group} + P_{setup})$, and P_{setup} can be derived from (24) with $W = W_{setup}$. P_{group} is the power consumed by all the modules in the group. Finally, we define the power conversion efficiency enhancement ($Gain_\eta$) and power loss reduction ($Gain_P$) by

$$Gain_\eta = \left(\frac{\eta_{opt}}{\eta_{def}} - 1\right) 100\ (\%)$$

$$Gain_P = \left(1 - \frac{P_{opt}}{P_{def}}\right) 100\ (\%). \qquad (25)$$

Table IV shows the S3 results for both patterns I and II, where the values of W_{opt} are $W_{opt,I}$ for pattern I, and $W_{opt,II}$ for pattern II. The overall power conversion efficiency enhancements for patterns I and II are 6% and 5.5%, which correspond to 19% and 18% power loss reductions during power conversion, respectively.

To check how much the voltage ripple increases by changing from Wdef,k to Wopt,k, we define a parameter called voltage ripple change (%) and calculated as $\Delta V_{opt,k}/\Delta V_{def,k} \cdot 100$, where $\Delta V_{def,k}$ and $V_{opt,k}$ are obtained by substituting $W_{def,k}$ and $W_{opt,k}$ in (6), respectively. Throughout the regression results and possible range of output current for each group, $V_{sw1} + V_{sw2} = r_1 I_{\text{out}}$, and $V_L = r_2 I_{\text{out}}$. Then, all the possible V_{sw1}, V_{sw2} and V_L are considered to derive the maximum voltage ripple change.

TABLE IV

STATIC SWITCH SIZING (S3) RESULTS (%) OF PATTERNS I AND II

k	$W_{opt,I}$	$Gain_\eta$	$Gain_P$	$W_{opt,II}$	$Gain_\eta$	$Gain_P$
1	0.2976	7.9718	29.7828	0.3365	6.4977	26.7376
2	0.1886	14.0471	38.7954	0.2073	12.3355	37.0320
3	0.3793	4.4908	11.4699	0.3869	3.9988	10.4214
4	0.1671	10.9028	29.7479	0.1834	9.7412	28.0067
5	0.1271	12.8754	27.7512	0.0900	12.4529	27.2171
6	0.1176	14.2869	34.0873	0.1228	13.6310	33.3380
7	0.2130	2.0874	11.0332	0.2145	2.0615	10.9291
t	—	6.0157	19.0699	—	5.5536	18.0396

*t indicates the overall result

TABLE V

VOLTAGE RIPPLE CHANGE (%) FROM $W_{def,k}$ TO $W_{opt,k}$ FOR PATTERNS I AND II

	$k = 1$	$k = 2$	$k = 3$	$k = 4$	$k = 5$	$k = 6$	$k = 7$
I	8.3	11.6	5.5	14.3	7.7	1.9	0.1
II	7.1	11.0	5.5	13.4	10.2	1.8	0.1

The results for each group are reported at Table V. Because the worst case is only 14%, and the equivalent converter model includes LDO, we can safely state that the resulting voltage ripple will satisfy the design constraints.

E. Simulation Results: Dynamic Switch Modulation

According to the classification flow in Fig. 10, Groups 1, 2, 3, and 5 are classified to discretizable, Groups 4 and 6 are discrete, and Group 7 belongs to continuous. In this section, only the results from the smartphone usage pattern, pattern I, is presented for brevity (i.e., the results from pattern II are almost same as the pattern I).

As a result of the K-means clustering procedure with $K = 7$ and $\lambda = 1000$, each of Groups 1, 2, 3 and 5 has seven discrete load current values. Table VI shows the resulted values and their corresponding optimum width values. From Algorithm III-B1, we can derive the set of switches of each group that covers the maximum number of the width values in Table VI. The boundary conditions of the load current region are calculated by $I_{bd,i} = \sqrt{W_i W_{i+1} r_3 / r_1}$, which is derived from (14).

Table VII shows examples of the resulted efficiency enhancement ($Gain_{\eta,method}$) of Groups 1, 2 and 3, when $\Delta = 0.4$, and $N = 3$ ($method = DSM1$) or 4 ($method = DSM2$). The results from the S3 ($method = S3$) are also provided for comparison. We assume that the power losses of the controller for all methods are the same. The table includes the results of the five applications. The results of the other five applications are omitted in this paper, but they show similar results to the application in the table. Rather, in order to demonstrate the effectiveness of DSM for the varying load conditions, three cases of (fixed) high load current conditions are also explored, although they are rarely observed when running the common applications.

Table VII shows that $Gain_{\eta,DSM2}$ is slightly better than $Gain_{\eta,DSM1}$, and $Gain_{\eta,DSM1}$ is generally better than $Gain_{\eta,S3}$. For Groups 1 and 2, high efficiency enhancement is achieved for all the methods when the applications require the low load current (i.e., System setting and Call in both groups, and

TABLE VI

RESULTS OF K-MEANS CLUSTERING FOR GROUPS 1, 2, 3 AND 5: $I'_{out,k}$ IS THE kTH MEAN VALUE (mA), AND $W_{opt,k}$ IS ITS CORRESPONDING OPTIMAL WIDTH

Group	$I'_{out,1}$	$W_{opt,1}$	$I'_{out,2}$	$W_{opt,2}$	$I'_{out,3}$	$W_{opt,3}$	$I'_{out,4}$	$W_{opt,4}$	$I'_{out,5}$	$W_{opt,5}$	$I'_{out,6}$	$W_{opt,6}$	$I'_{out,7}$	$W_{opt,7}$
1	26	0.1509	39	0.2263	52	0.3017	69	0.4004	98	0.5716	199	1.1547	300	1.7408
2	21	0.1088	28	0.1451	37	0.1918	48	0.2514	82	0.4250	188	0.9744	296	1.5341
3	117	0.2970	138	0.3489	147	0.3717	156	0.3945	163	0.4122	258	0.6540	354	0.8983
5	16	0.0968	17	0.1068	19	0.1155	20	0.1231	22	0.1350	25	0.1533	28	0.1719

TABLE VII

EFFICIENCY ENHANCEMENT RESULTS (%) OF THE DYNAMIC SWITCH MODULATION ($Gain_{\eta,DSM}$) AND THE STATIC SWITCH SIZING ($Gain_{\eta,S3}$)

Group 1 - $DSM1$: $N=3$, Width set={0.1509, 0.4404, 1.1547} and $DSM2$: $N=4$, Width set={0.1509, 0.3017, 0.4404, 1.1547}							
Case	$Gain_{\eta,S3}$	$Gain_{\eta,DSM1}$	$Gain_{\eta,DSM2}$	Case	$Gain_{\eta,S3}$	$Gain_{\eta,DSM1}$	$Gain_{\eta,DSM2}$
System setting	9.7606	10.0825	10.2681	Neocore	3.9914	4.4510	4.4730
Call	8.7008	8.6859	8.9136	$I_{out}=150mA$	-3.7896	0.1598	0.4380
Skype-videochat	3.8615	4.4978	4.5696	$I_{out}=250mA$	-9.5422	0.0685	0.0851
Facebook	3.9612	4.5575	4.6576	$I_{out}=350mA$	-13.7721	0.7423	0.8465
Group 2 - $DSM1$: $N=3$, Width set={0.1088, 0.4250, 0.9744} and $DSM2$: $N=4$, Width set={0.1088, 0.1918, 0.2514, 0.9744}							
Case	$Gain_{\eta,S3}$	$Gain_{\eta,DSM1}$	$Gain_{\eta,DSM2}$	Case	$Gain_{\eta,S3}$	$Gain_{\eta,DSM1}$	$Gain_{\eta,DSM2}$
System setting	10.0681	11.7873	12.2481	Neocore	12.1440	11.2321	12.2433
Call	12.9531	12.3106	13.1478	$I_{out}=150mA$	-6.4464	0.5639	0.5639
Skype-videochat	1.8863	4.2109	4.2728	$I_{out}=250mA$	-13.9895	-0.1039	0.0045
Facebook	7.8378	8.2429	8.7913	$I_{out}=350mA$	-19.5088	0.3518	2.4955
Group 3 - $DSM1$: $N=3$, Width set={0.2970, 0.3945, 0.8983} and $DSM2$: $N=4$, Width set={0.2970, 0.3717, 0.4122, 0.8983}							
Case	$Gain_{\eta,S3}$	$Gain_{\eta,DSM1}$	$Gain_{\eta,DSM2}$	Case	$Gain_{\eta,S3}$	$Gain_{\eta,DSM1}$	$Gain_{\eta,DSM2}$
System setting	4.4993	4.5576	4.6029	Neocore	-3.0773	0.4821	0.4636
Call	4.8733	4.9012	4.9499	$I_{out}=200mA$	2.7330	2.8415	8.3743
Skype-videochat	-0.0035	1.5326	1.5531	$I_{out}=250mA$	0.7476	4.7393	4.8130
Facebook	3.2841	3.4753	3.5499	$I_{out}=300mA$	-5.8153	2.1294	2.2558

Facebook and Neocore in Group 2), which are around 8% to 12%. On the other hand, for the applications requiring the higher load current (i.e., Skype-videochat in both group, and Facebook and Neocore in Group 1), the efficiency enhancement of Groups 1 and 2 are not that high, which is around 1% to 4%. That is because the efficiencies from the default setup (η_{def}) are higher in the high load current conditions than the low load current conditions. Meanwhile, the S3 achieves high efficiency enhancement at the low load current conditions, as shown in Table VII. But it has drawbacks that the efficiencies at the high load current conditions are reduced—$Gain_{\eta,S3}$ can be even negative. On the other hand, DSM can achieve the high efficiency enhancement for wide load current range. For example, the result of Skype-videochat at Group 2, $Gain_{\eta,S3}$ is 1.8%, but $Gain_{\eta,DSM1}$ and $Gain_{\eta,DSM2}$ are still more than 4%. Furthermore, in the cases of Groups 1 and 2 when the load current conditions are 150, 250, and 350 mA, the results demonstrate that DSM keeps efficiency enhancement even for the high current region, but the S3 does not. The results from Group 3 in Table VII show the similar results.

The K-means clustering result for Group 5 in Table VI shows the gap between minimum and maximum load current conditions is only 12 mA. Thus, only one switch set (N = 1) sized by the S3 would be enough. For DMS with N = 2, {0.0968, 0.1231} can be a set of widths of the switches.

Camera-digital in Group 4 is the module that has the on/off operation controlled by an user. Furthermore, as shown in Fig. 15, it dominantly consumes power (55%–65%). When the camera is on, the average load current of Group 4 is 62.7269 mA, and when it is off, the average load current of Group 4 is 19.5208 mA. From (23), these current values correspond to the width of switches as 0.5127 and 0.1595, respectively. Meanwhile, SD card and Camera analog are such modules in Group 6. Then, we have four discrete load current values, 15, 29, 37, and 51 (mA), according to the conditions of (SD card, Camera analog): off/off, off/on, on/off and on/on. These current values correspond to 0.1128, 0.2181, 0.2783, and 0.3836 as the optimum effective widths, respectively. When N = 2, {0.1128, 0.2783} can be a set of widths of the switches. Table VIII shows the efficiency enhancement

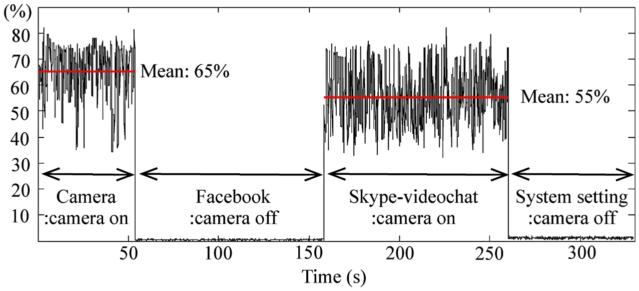


Fig. 15. Ratio of the power consumed by camera digital to the power consumed by all the modules in Group 4.

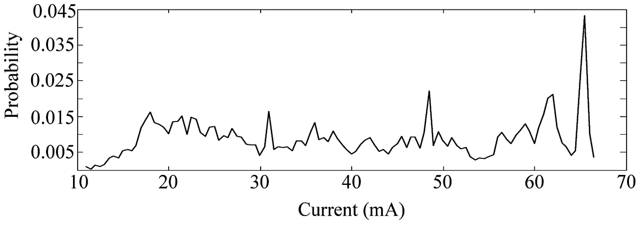


Fig. 16. Load current distribution of display modules in Group 7 according to the ten brightness levels.

TABLE VIII
EFFICIENCY ENHANCEMENT RESULTS OF GROUPS 4 AND 6

	Group 4		Group 6	
Case	$Gain_{\eta,S3}$	$Gain_{\eta,DSM}$	$Gain_{\eta,S3}$	$Gain_{\eta,DSM}$
System setting	16.7919	16.8653	12.9967	12.9725
Neocore	5.6650	5.6295	13.7939	13.7870
Skype-videochat	-1.4018	1.6159	5.3388	6.0069
Camera	0.1997	2.3499	6.0982	6.6365

TABLE IX
RESULTS OF THE POWER LOSS GAIN (%) FOR TEN APPLICATIONS

Application	$Gain_{P,S3}$	$Gain_{P,DSM1}$	$Gain_{P,DSM2}$
Call	18.3237	18.8361	19.0967
Camera	15.1582	15.9335	16.1485
Clock	22.4631	23.3820	23.4811
Facebook	17.2478	18.3995	18.6534
GoogleMap	16.6835	18.3276	18.5324
Neocore	4.8926	11.1143	11.1352
Skype-videochat	9.5858	12.2300	12.2963
SMS	18.4505	19.4249	19.6605
System setting	21.0103	21.5473	21.6712
Youtube	17.7074	18.2816	18.5089

results of Group 4 and 6, for the four applications. The case of Group 4 shows the similar results to the previous cases of Groups 1, 2, and 3 that DSM performs as well as the S3 does in the low load current conditions (i.e., System setting and Neocore), but DSM also keeps positive enhancements even in the high load current conditions (i.e., Skype-videochat and Camera). On the other hand, the case of Group 6 shows that both methods have almost same results. That is because the load current range of Group 6 is narrow, besides the applications may not frequently require the maximum current.

Group 7 consists of two modules, display memory and display backlight. Display backlight has various brightness levels that can be set by the user preference. We divide the brightness levels by 10, and measure the load current of Group 6 for each level. Then, the load current condition induced by each bright level is overlapped to the conditions of the adjacent levels. Fig. 16 shows the resulted load current distribution of Group 6, when all the levels are equally likely to occur. Next, we select the seven discrete current values of an arithmetic sequence satisfying that the minimum and maximum current values are 11 and 66 mA, respectively. These current values corresponds to the required width values. From Algorithm 1 with $\Delta = 0.01$, a set, $\{0.0355, 0.1225, 0.2128\}$, is derived. All the possible effective width from the set can cover the seven required width values (thus $N = 3$ is enough in this case). Finally, we have the enhancement results that $Gain_{\eta,S3} = 3.9483\%$, and $Gain_{\eta,DSM} = 4.3424\%$, in the case of all the levels to be equally likely chosen.

For interested readers, we also provide Table IX to show the detailed results for the ten applications.

VI. CONCLUSION

This paper demonstrated that significant power loss occurs during power conversion in the PDN of a smartphone platform. To mitigate this problem, this paper focuses on the dc–dc converters in the PDN to introduce two optimization methods for the dc–dc converters. S3 was presented to configure the switches in dc–dc converters so that the optimal operating conditions of the dc–dc converters match to the general load current conditions. The general load current distributions for all modules in the platform were derived from the measured loading profiles and smartphone usage patterns. DSM was also presented to overcome the lack of capability of the S3 that may not be optimal for dynamically varying load conditions. By exploiting the multiswitching scheme, detailed procedures to select and size the switches were introduced. To verify the presented methods in an actual smartphone platform, the PDN characterization procedure was performed. By the proposed equivalent converter model and grouping method, the power conversion efficiency of the PDN in the target smartphone platform could be characterized. Finally, we applied the proposed optimization methods to the platform. The experimental results showed that the S3 achieves 6% overall efficiency enhancement, which translates to 19% power loss reduction for the general smartphone usage pattern. The DSM accomplishes the similar improvement at the same condition. Furthermore, it also can achieve the high efficiency enhancement in the various load conditions. In the design flow, both S3 and DSM methods can be applied only after obtaining the load current distributions for the modules. S3 is simple to implement, but may not produce the optimal transistor widths under dynamically changing load conditions or even under the case that the load distribution has a high variance. On the other hand, DSM has more control/area overhead than S3, but it can achieve high conversion efficiency enhancement under all load conditions. Note that if it happens that the load current distributions are changed because of newly added applications or changing usage patterns compared to those used for the initial optimization, the DSM method will continue to provide power efficiency enhancement because of its adaptability whereas the S3 method will fail.

REFERENCES

[1] W. Lee, Y. Wang, D. Shin, N. Chang, and M. Pedram, "Power conversion efficiency characterization and optimization for smartphones," in *Proc. Int. Symp. Low Power Electron. Design*, 2012, pp 103–108.

[2] L. Zhang, B. Tiwana, R. P. Dick, Q. Zhiyun, Z. M. Mao, W. Zhaoguang, and Y. Lei, "Accurate online power estimation and automatic battery behavior based power model generation for smartphones," in *Proc. Int. Conf. Hardware/Software Codesign Syst. Synthesis*, 2010, pp. 105–114.

[3] M. Dong and L. Zhong, "Self-constructive high-rate system energy modeling for battery-powered mobile systems," in *Proc. Int. Conf. Mobile Syst. Appl. Services*, 2011, pp. 335–348.

[4] A. Pathak, Y. C. Hu, and M. Zhang, "Fine-grained energy accounting on smartphones with Eprof," in *Proc. EuroSys*, 2011, pp. 29–42.

[5] D. Shin, N. Chang, W. Lee, Y. Wang, Q. Xie, and M. Pedram, "Online estimation of the remaining energy capacity in mobile systems considering system-wide power consumption and battery characteristics," in *Proc. Asia South Pacific Des. Autom. Conf.*, 2013, pp. 59–64.

[6] L. Benini, A. Bogliolo, and G. D. Micheli, "A survey of design techniques for system-level dynamic power management," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 8, no. 3, pp. 299–316, Jun. 2000.

[7] Q. Qiu, Q. Wu, and M. Pedram, "Dynamic power management in a mobile multimedia system with guaranteed quality-of-service," in *Proc. Design Autom. Conf.*, 2001, pp. 834–839.

[8] Y. Choi, N. Chang, and T. Kim, "DC-DC converter-aware power management for low-power embedded systems," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 26, no. 8, pp. 1367–1381, Aug. 2007.

[9] B. Amelifard and M. Pedram, "Optimal design of the power-delivery network for multiple voltage-island system-on-chips," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 28, no. 6, pp. 888–900, Jun. 2009.

[10] Z. Zeng, X. Ye, Z. Feng, and P. Li, "Tradeoff analysis and optimization of power delivery networks with on-chip voltage regulation," in *Proc. Design Autom. Conf.*, 2010, pp. 831–836.

[11] O. Abdel-Rahman, J. A. Abu-Qahouq, L. Huang, and I. Batarseh, "Analysis and design of voltage regulator with adaptive FET modulation scheme and improved efficiency," *IEEE Trans. Power Electron.*, vol. 23, no. 2, pp. 896–906, Mar. 2008.

[12] S. Kudva and R. Harjani, "Fully-integrated on-chip DC-DC converter with a 450X output range," *IEEE J. Solid-State Circuits*, vol. 46, no. 8, pp. 1940–1951, Aug. 2011.

[13] A. A. Sinkar, H. Wang, and N. S. Kim, "Workload-aware voltage regulator optimization for power efficient multicore processors," in *Proc. Design Autom. Test Eur.*, 2012, pp. 1134–1137.

[14] J. Wibben and R. Harjani, "A high-efficiency DC-DC converter using 2nH integrated inductors," *IEEE J. Solid-State Circuits*, vol. 43, no. 4, pp. 844–854, Apr. 2008.

[15] R. Erickson and D. Maksimovic, *Fundementals of Power Electronics*. Berlin, Germany: Springer, 2001.

[16] J. Xiao, A. Peterchev, J. Zhang, and S. Sanders, "An ultra-low-power digitally-controlled buck converter IC for cellular phone applications," in *Proc. Appl. Power Electron. Conf.*, 2004, pp. 383–391.

[17] C. Shi, B. C. Walker, E. Zeisel, E. B. Hu, and G. H. McAllister, "A highly integrated power management IC for advanced mobile applications," in *Proc. Custom Integr. Circuits Conf.*, 2006, pp. 85–88.

[18] Y. Du, M. Wang, R. T. Meitl, S. Lukic, and A. Q. Huang, "High-frequency high-efficiency DC-DC converter for distributed energy storage modularization," in *Proc. Int. Experts Consultants*, 2010, pp. 1832–1837.

[19] S. Musunuri and P. L. Chapman, "Optimization of CMOS transistors for low power dc-dc converters," in *Proc. Power Electron. Specialist Conf.*, 2005, pp. 2151–2157

[20] A. Shye, B. Scholbrock, and G. Memik, "Into the wild: Studying real user activity patterns to guide power optimizations for mobile architectures," in *Proc. Int. Symp. Microarchitecture*, 2009, pp. 168–178.

[21] F. Hossein, M. Ratul, K. Srikanth, L. Dimitrios, G. Ramesh, and E. Deborah, "Diversity in smartphone usage," in *Proc. Int. Conf. Mobile Syst. Appl, Services*, 2010, pp. 179–194.

[22] T. M. T. Do, J. Blom, and D. Gatica-Perez, "Smartphone usage in the wild: A large-scale analysis of applications and context," in *Proc. Int. Conf. Multimodal Interaction*, 2011, pp. 353–360.

[23] PTM [Online]. Available: http://ptm.asu.edu (accessed 2013 Jun.)

[24] Texas Instruments. *Handset: Smartphone Solutions* [Online]. Available: http://www.ti.com/solution/handset_smartphone (accessed 2013 Jun.)

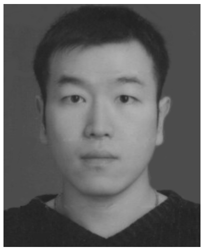
[25] Qualcomm. *Snapdragon MDP MSM8660 Datasheet* [Online]. Available: https://developer.qualcomm.com/mobile-development/development-devices/snapdragon-mdp-legacy-devices (accessed 2013 Jun.)

[26] Antutu [Online]. Available: http://www.antutu.net (accessed 2013 Jun.)

[27] Vellamo [Online]. Available: http://www.quicinc.com/vellamo (accessed 2013 Jun.)

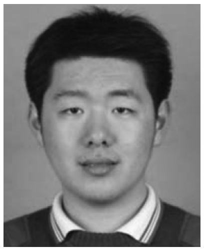
[28] Quadrant [Online]. Available: http://www.aurorasoftworks.com (accessed 2013 Jun.)

[29] GLBenchmark [Online]. Available: http://gfxbench.com (accessed 2013 Jun.)



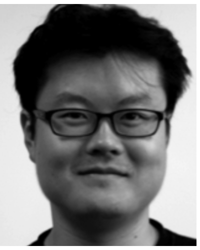
Woojoo Lee (S'12) received the B.S. degree in electrical engineering from Seoul National University, Seoul, Korea, in 2007, and the M.S. degree in electrical engineering from the University of Southern California, Los Angeles, CA, USA, in 2010. He is currently pursuing the Ph.D. degree in electrical engineering at the Department of Electrical and Electronic Engineering, University of Southern California, under the supervision of Prof. M. Pedram.

His current research interests include low-power VLSI design, system-level power management, and embedded system designs.



Yanzhi Wang (S'12) received the B.S. degree with distinction in electronic engineering from Tsinghua University, Beijing, China, in 2009. He is currently pursuing the Ph.D. degree in electrical engineering at the Department of Electrical and Electronic Engineering, University of Southern California, Los Angeles, CA, USA, under the supervision of Prof. M. Pedram.

His current research interests include system-level power management, next-generation energy sources, hybrid electrical energy storage systems, near-threshold computing, and the smart grid. He has published around 60 papers in these areas.



Donghwa Shin (S'05–M'12) received the B.S. degree in computer engineering and the M.S. and Ph.D. degrees in computer science and electrical engineering from Seoul National University, Seoul, Korea, in 2005, 2007, and 2012, respectively.

He is currently with the Dipartimento di Automatica e Informatica—EDA Group, Politecnico di Torino, Torino, Italy, as a Research Assistant. His current research interests include system-level low-power techniques for embedded systems and hybrid power system design for embedded systems.

Naehyuck Chang (F'12) received the B.S., M.S., and Ph.D. degrees from the Department of Control and Instrumentation, Seoul National University, Seoul, Korea, in 1989, 1992, and 1996, respectively.

He joined the Department of Computer Engineering, Seoul National University, in 1997, where he is currently a Professor with the Department of Electrical Engineering and Computer Science and is the Vice Dean of the College of Engineering. His current research interests include low-power embedded systems, hybrid electrical energy storage systems, and next-generation energy sources.

Dr. Chang has served on technical program committees in many EDA conferences, including DAC, ICCAD, ISLPED, DATE, CODES+ISSS, and ASP-DAC. He was a TPC Chair or Co-Chair of RTCSA 2007, ISLPED 2009, ESTIMedia 2009 and 2010, and CODES+ISSS 2012, and will serve as the TPC Chair of ICCD 2014, and ASP-DAC 2015. He was the General Vice Chair of ISLPED 2010, and the General Chair of ISLPED 2011 and ESTIMedia 2011. He has served as an Associate Editor of IEEE TCAS-I, IEEE TCAD, ACM TODAES, and ACM TECS, Springer DAES, and was a Guest Editor of ACM TODAES in 2010, and ACM TECS in 2010 and 2011. He is the ACM SIGDA Chair and an ACM Distinguished Scientist.

Massoud Pedram (F'01) received the Ph.D. degree in electrical engineering and computer sciences from the University of California, Berkeley, CA, USA, in 1991.

He is the Stephen and Etta Varra Professor with the Ming Hsieh Department of Electrical Engineering, University of Southern California, Los Angeles, CA, USA. He holds ten U.S. patents and has published four books, 12 book chapters, and more than 130 archival and 320 conference papers. His current research interests include low power electronics, energy-efficient processing, and cloud computing to photovoltaic cell power generation, energy storage, and power conversion, and from RT-level optimization of VLSI circuits to synthesis and physical design of quantum circuits.

Dr. Pedram was a recipient of the 1996 Presidential Early Career Award for Scientists and Engineers, an ACM Distinguished Scientist, and currently serves as the Editor-in-Chief of the *ACM Transactions on Design Automation of Electronic Systems* and the IEEE JOURNAL ON EMERGING AND SELECTED TOPICS IN CIRCUITS AND SYSTEMS. For this research, he and his students have received six conference and two IEEE Transactions Best Paper Awards. He has also served on the Technical Program Committee of a number of premiere conferences in his field and was the Founding Technical Program Co-Chair of the 1996 International Symposium on Low Power Electronics and Design and the Technical Program Chair of the 2002 International Symposium on Physical Design.