Optimizing a Reconfigurable Power Distribution Network in a Multicore Platform

Woojoo Lee, *Student Member, IEEE,* Yanzhi Wang, *Student Member, IEEE,*
and Massoud Pedram, *Fellow, IEEE*

***Abstract*—The emerging trend towards utilizing chip multicore processors (CMPs) that support dynamic voltage and frequency scaling (DVFS) is driven by user requirements for high performance and low power. To overcome limitations of the conventional chip-wide DVFS and achieve the maximum possible energy saving, per-core DVFS is being enabled in the recent CMP offerings. While power consumed by the CMP is reduced by per-core DVFS, power dissipated by the set of voltage regulators (VRs) that are required to support per-core DVFS becomes critical. This paper focuses on the dynamic control of the VRs in a CMP platform. Starting with a proposed platform with a reconfigurable VR-to-core power distribution network (PDN), two optimization methods are presented to maximize the system-wide energy savings: (i) reactive VR consolidation to reconfigure the network for maximizing the power conversion efficiency of the VRs, which is performed under the pre-determined DVFS levels for the cores, and (ii) proactive VR consolidation to determine new DVFS levels for maximizing the total energy savings without any performance degradation. Along with the optimization methods for the PDN composed of homogeneous VRs, we also discuss the PDN with heterogeneous VRs, which is proposed to increase the benefits of the VR consolidation by incorporating VRs with a larger driving capability of load current. Results from detailed simulations based on realistic experimental setups demonstrate up to 36% VR energy loss reduction and 9% total energy saving.**

Keywords Low-power design; Power distribution network; Power delivery network; Voltage regulator; DC-DC converter;

I. Introduction

By leveraging technology scaling to pack multiple processor cores on a single die, chip multi-core processors (CMPs) have been increasingly adopted in desktop and server applications, as well as mobile environments, due to the growing demand for high performance VLSI systems. CMPs have achieved high throughputs in handling multiple applications by distributing them to different cores and executing them simultaneously. Furthermore, emerging challenging scientific and engineering problems craving for high performance computing and simulation have resulted in the advent of many-core processors. In spite of the benefits, developing such multi/many-core processors has hit a critical roadblock, power consumption. Due to the limited power budget and running/cooling cost, power consumption has become a overriding concern for CMP designs.

One of the most effective techniques to mitigate the power consumption of CMPs is to dynamically vary the supply voltage and operating frequency values applied to the process cores in response to load conditions or workload characteristics (this is known as dynamic voltage and frequency scaling, or DVFS for short) [2], [3]. The conventional approach is to perform DVFS for all cores in a processor (per-chip DVFS). This approach has not been able to take full advantage of power-saving that DVFS potentially achieves. For instance, some of the cores may not need a high voltage/frequency level, but can not be lowered because of the other cores. To surmount this shortcoming, applying DVFS to each individual core (per-core DVFS) or to the clustered cores (per-cluster DVFS) has been suggested, and has resulted in excellent flexibility in controlling power [4], [5], [6]. Unfortunately, this approach can still have inevitable drawbacks such as a larger footprint, higher power conversion loss, and/or higher control complexity incurred by the more sophisticated power distribution network (PDN).

The PDN in the per-core DVFS platform provides power to each core from a power source. It consists of voltage regulators (VRs), which play a pivotal role to convert the voltage level of the power source to the required voltage levels of the target cores. Therefore, to support the per-core DVFS, at least the same number of VRs (as the number of cores) should be equipped in the platform, which can cause high area overhead. However, recent research work that focuses on on-chip VR designs proves that this overhead can be significantly mitigated by reducing the size of each VR [7], [8], [9].

Meanwhile, the VRs inevitably dissipate power, and power dissipations from all VRs inside a per-core DVFS platform can result in a considerable amount of power loss. Given that a VR's power conversion efficiency (simply called VR efficiency in the remainder of paper) is the ratio of the power consumed by a core to the total power consumed by both the core and VR, the state-of-the-art VRs exhibit high peak power conversion efficiency, but their efficiency can drop dramatically under adverse load conditions (i.e., out-of-range output current levels) [8], [1]. Fig. 1 shows an example of traces of the VR efficiency during delivering power to a core. Around 24% of input power is dissipated by the VR in the high efficiency region (indicated by the red line), but more than 53% of the input power is consumed by the VR in the low effi-

This research is sponsored by grants from the Semiconductor Research Corporation, the National Science Foundation, and the Defense Advanced Research Projects Agency. This paper was presented in part at the 2014 Design, Automation and Test in Europe (DATE), Dresden, Germany [1]. This paper was recommended by Associate Editor T. Kim.

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

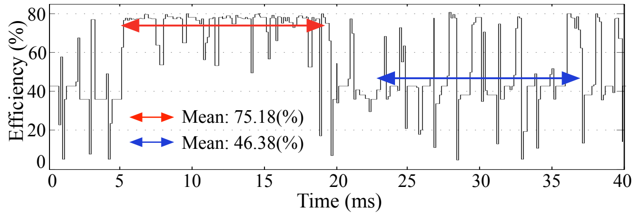


Fig. 1. Power conversion efficiency traces: simulation result from Parsec-*Streamcluster* in Sniper [10] with LTC3816 [11].

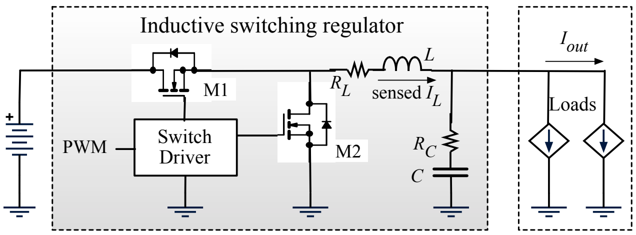


Fig. 2. Circuit schematics of a inductive (synchronous) switching VR.

ciency region (the blue line) in the figure. Consequently, the VR efficiency is a critical concern and optimization objective to save power in the platform.

A few recent papers have studied VR components in order to improve the efficiency of a single VR [12], [13], [14], [15]. Optimizing the switch sizes and the frequency of the pulse-width modulator (PWM) in the VR for the given workload has been studied in [13], [14]. Using multiple/parallel switches in the VR design has been presented in [12], [15]. In contrast, little attention has been paid to the question of how to improve the efficiency of a VR network from system-level optimizations, in spite of a few papers that have explored VRs from a system perspective [16], [4], [5], [6]. A DVFS policy that is aware of the VR efficiency characteristics has been addressed in [16]. The optimal frequency of a core has been derived to minimize the total energy consumption in both the core and the VR. However, there is still large potential to save more power in the multi-core and multi-VR systems. In [4], the potential of energy saving in the CMP using per-core DVFS and fast transient responses of VRs has been presented. To determine the optimal DVFS levels for each core, an offline algorithm based on *integer linear programming* (ILP) has been proposed. But this approach does not consider the power dissipated by the indispensable large number of VRs to enable per-core DVFS. Meanwhile, to tackle the drawback of per-core DVFS, an offline approach to cluster the cores in the same voltage-rail has been suggested [5]. *K-means clustering* has been used to group some cores which have the similar DVFS levels, so as to reduce the number of VRs required in the system. However, reducing a fixed number of VRs loses in part the benefit of per-core DVFS as aforesaid, and may not guarantee energy saving in VRs with dynamically changing workloads. In addition, clustering the cores with similar behaviors of the voltage/frequency levels may not be applicable for multi-threaded applications where the locking and synchronization issues should be carefully accounted for [17], [18]. For example, a delayed thread of an application on the clustered core may have to lock the other threads for the synchronization, which can cause significant delay of the application. Similar to [5], but to support an online control of matching cores and converters, a (3D) reconfigurable switch network has been recently presented in [6]. This approach has achieved to reduce the number of VRs, and flexibly utilize them with a proposed time-space multiplexing scheme. However, platform-level total power consumption that should include power consumption of the multiple VRs has not been taken into account.

This paper starts from a concept to combine some cores, which operate at the same voltage level and drive relatively small amount of load current, to be powered by a single VR. This approach can significantly reduce the VR power loss in the multi-core processor platform due to the following two reasons: (i) the VR used to power multiple cores has relatively high current load and thus has higher efficiency according to the VR characteristics, and (ii) the VRs that are not used can be turned off to save power. Based on this concept of VR consolidation, we propose a new design of the multi-core platform, which exploits (multiple) sets of network switches to reconfigure the PDN. We then present two optimization methods to minimize the VR power loss and maximize the total energy saving. We first propose a reactive method that configures the PDN based on the sensed voltage/current level of each core. We present a proactive method to decide the optimal voltage/frequency level of each core in the consideration of maximizing the consolidation opportunities of VRs, in order to minimize the energy consumption of the whole system. Along with the optimization methods for the PDN composed of homogeneous VRs, we also discuss the PDN with heterogeneous VRs, which is proposed to increase the benefits of the VR consolidation by equipping VRs with a larger driving capability of load current. We provide detailed discussion about the design considerations for both homo/heteogeneous PDNs.

We validate the proposed methods on various applications from the PARSEC [19] and SPLASH2 [20] benchmark suites. We perform detailed multi-core processor simulation using the modified Sniper simulator [10], and the spice circuit simulation with a commercial VR carefully selected for fair evaluation. Results demonstrate up to 36% VR energy loss reduction and 9% total energy saving.

The remainder of this paper is organized as follows. Section II provides some characteristics of the VR model. In Section III, the two optimization methods are presented. Section IV introduces the PDN with the heterogenous VRs, and extends the two optimization methods. Section V is dedicated to the experimental work, while Section VI concludes the paper.

II. Preliminary: VR characteristics

According to circuit implementation and operation principles, voltage regulators can be generally classified into three types, low-dropout regulators (LDOs), switched-capacitor regulators (SCs) and inductive switching regulators. LDOs and SCs have advantages that they are easy for integration and

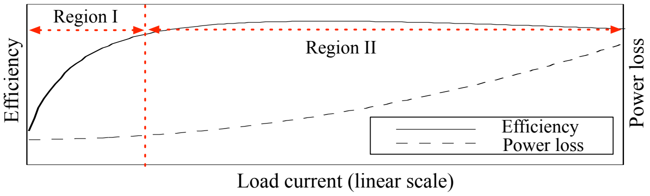


Fig. 3. VR efficiency and power loss vs. output current conditions.

have low area-overhead compared to inductive switching regulators. However, inductive switching regulators achieve higher conversion efficiencies over a wide range of output loads. Furthermore, the digitally programable controllers equipped in inductive switching regulators have more benefits than other types of regulators to support dynamic voltage setting with fast transient response. Therefore, inductive switching regulators are more suitable and typically used for delivering power to processors. We focus on the inductive switching regulator, and simply call it VR in the remainder of this paper.

To help understanding the power losses of the VR, a simplified schematics of a synchronous buck-type VR is shown at Fig. 2. This schematic consists of an inductor, a capacitor, two switches, and a switch driver which is a pulse-width-modulation (PWM) controller. If the conduction current of the switch is small (e.g., less than 0.5A [21]), the lateral power MOSFETs have been used for the two switches [16], [12], [13], [15]. Whereas, the trench (vertical) power MOSFETs are widely used in state-of-the-art VRs, because they generally offer much lower resistance than the lateral MOSFETs. In this paper, we uses the trench power MOSFETs for the two switches, in order to follow the trend of the modern VR designs that are dominantly equipped in the multicore platforms.

In Fig. 2, M1 and M2 are the high side control FET and low side synchronous FET, respectively. Parasitic resistance of the inductor L is denoted by R_L. In the same manner, the parasitic resistance of the capacitor C is referred to R_C. Depending on the physical sources of power consumption, power loss of the VR is composed of the following three parts: conduction loss, switching loss, and controller power loss, denoted by $P_{conduction}$, $P_{switching}$ and $P_{controller}$, respectively [16], [15]. The power loss of the VR, P_{loss}, is the sum of the three parts:

$$\begin{aligned} P_{loss} =& I_{out}{}^2(R_L + DR_{M1} + (1-D)R_{M2}) \\ &+ (\Delta I)^2(R_L + DR_{M1} + (1-D)R_{M2} + R_C)/12 \\ &+ V_{in} f_{sw}(Q_{M1} + Q_{M2}) + V_{in} I_{controller}, \end{aligned} \tag{1}$$

where I_{out} is the output current and, V_{in} and V_{out} are the input and output voltages; R_{M1} and R_{M2} are the resistance of the switch M1 and M2, respectively; Q_{M1} and Q_{M2} are the charge of each switch that includes gate charge, gate-source charge, output charge, and the diode reverse recovery charge [22]; D is the PWM duty ratios of the control FET, which can be derived from $\frac{V_{out} + I_{out}(R_{M2} + R_L)}{V_{in} - I_{out}(R_{M1} + R_{M2})}$; f_{sw} is the PWM switching frequency; $I_{controller}$ is the current flowing in the controller of the VR, and ΔI is the inductor current ripple. In (1), the first and second terms are the DC and AC parts of $P_{conduction}$, respectively. The third term of (1) is $P_{switching}$. The fourth term of (1) is $P_{controller}$. Finally, the VR efficiency, η, can be calculated as:

$$\eta(\%) = \frac{P_{out}}{P_{in}} = \frac{V_{out} I_{out}}{V_{out} I_{out} + P_{loss}} \cdot 100\% \tag{2}$$

As seen from (1), both resistances and charges of the switches contribute to the VR power loss, P_{loss}. While the resistances of the switches traditionally dominated P_{loss} at low f_{sw}, the charges of the switches has become more dominant as f_{sw} has been raised up to megahertz. Furthermore, exploiting the trench power MOSFETs that can offer very low resistances could reduce the conduction loss of the switches, $P_{conduction}$, but nevertheless the trench power MOSFETs still suffer from high charges due to the inherent vertical structure. Although there have been studies to overcome this drawback by optimizing the size of the switches [13], [14] and using multiple parallel switches [12], [15], these studies could not make the VR efficiency constantly high in the whole range of output current conditions. Instead, there will still exist low efficiency regions for the output current conditions, where the switching loss, $P_{switching}$, is dominant. For better understanding, Fig. 3 is provided to show an example of the VR efficiency according to the output current changes, based on (1). The output currents in the figure are conceptually divided to two regions to show that the main sources of the VR power loss are $P_{switching}$ and $P_{controller}$ in Region I, and $P_{conduction}$ in Region II. While Regions II shows relatively high efficiency owing to the low resistances of the switches, the efficiency in Region I drops dramatically under the adverse conditions of the output current due to the power loss from the high charges of the switches.

Selecting the pertinent switches are not only critical for the VR efficiency, but also affect the output current driving capability of the VR. Because (i) there exists a maximum (continuous) drain current for the switch, I_D, above which it may break down or get overheated, and (ii) the drain current is proportional to the output current, and the maximum output current $I_{out(max)}$ should be limited according to the required limitation of the inductor current ripple, ΔI_L. For reference, $I_{out(max)}$ may be expressed as [23]:

$$I_{out(max)} < I_{L(peak)} - \frac{\Delta I_L}{2} = I_{L(peak)} - \frac{V_{out}}{2 f_{sw} L}\left(1 - \frac{V_{out}}{V_{in}}\right), \tag{3}$$

where $I_{L(peak)}$ is the peak inductor current. In general, modern VRs feature an onboard sensing circuit that senses $I_{L(peak)}$, and a feedback control loop to limit the current.

In this paper, we carefully select the power MOSFETs from the device industry so that each single VR has the pertinent characteristics (efficiency and output current driving capability). Details will be discussed in Section V-A. Then we perform the system level optimizations for the whole VRs in the power distribution network of a multicore platform.

III. Dynamic Reconfiguration of the VR-to-core Network

A state-of-the-art VR powering a set of cores may have low conversion efficiency when there is a mismatch between

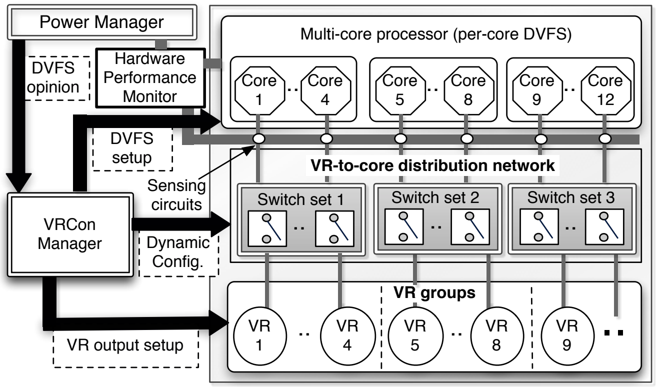


Fig. 4. Diagram of the proposed multicore platform.

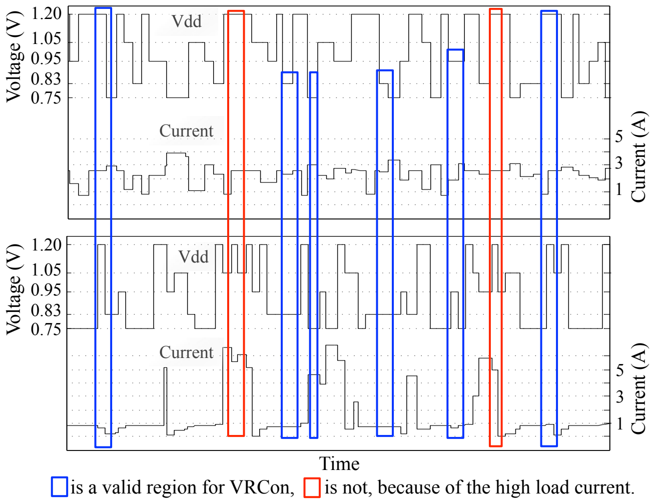


Fig. 5. Example cases that the reactive VRCon can be applied.

the high efficiency region of VRs and the load condition of the cores, as addressed in the previous section. Furthermore, due to the introduction of a large number of VRs for per-core DVFS, significant amount of power will be dissipated by the VRs.

Especially, the VR efficiency under the low load current condition, as shown in Region I of Fig. 3, could not be effectively improved by the approaches of sizing the switches. In addition, the power consumption by the controller in a VR, $P_{controller}$, cannot be scaled with the size of switches. In Region I where the PWM operating mode is inefficient, an alternative operating mode such as pulse frequency modulation (PFM) can be added to compensate the degraded efficiency [8], [12]. Although mitigating the radical efficiency drop in the low current region, the efficiency of the PFM mode is typically lower than that of the PWM mode in the normal current region. The design/control complexity of the VR also increases by supporting switching between these two modes.

Instead of adding more operating modes, we propose a system-level optimization technique to substantially improve the VR efficiency in the per-core DVFS based CMPs. This technique dynamically configures the connection network between VRs and cores according to the load current demand for each core. The basic idea can be motivated and illustrated with a simple example: if both cores in a dual core processor require the same supply voltage level, and they have small load currents (their load currents are not necessarily the same), then their power domains can be consolidated to share a single VR. In this way, the shared VR will have higher load current and thus higher conversion efficiency (because it will subsequently operate in its high conversion efficiency region), whereas the other VR which is not in use can be turned off to save energy. Starting from this intuition, we propose a new technique called VR consolidation (or VRCon for short) in a reconfigurable VR-to-core distribution network (this is in analogy with the well-known technique of core consolidation used to consolidate tasks/jobs into a minimum number of active cores in a CMP).

A. Proposed multicore platform

Fig. 4 provides a conceptual diagram of the proposed multicore platform. The platform has a number of VRs and multiple cores. There are several groups of reconfigurable VR-to-core connection networks supported by network switches implemented with power MOSFET switches. The VR-to-core network can deliver power for each core from any VR in the same group. We will discuss these groups of connection networks in detail in Section III-D. This reconfigurable power distribution network thus enables arbitrary connections between output of any VR and the input power pin of any core in the same group.

The power manager (PM) in a conventional CMP platform controls the processor's operating condition by using the DVFS technique. Compared to the conventional designs, we add a VRCon manager (called VRCM), which ultimately controls the core's frequency/voltage level, as well as the operations of VRs and ON/OFF states of the network switches in VRCon. The PM in the proposed platform still keeps monitoring the core status (i.e., performance) reported by the hardware performance monitor (HPM) as a conventional PM does. According to this design, the PM determines a tentative supply voltage and operating frequency of each core, and transmits this information to VRCM as a recommendation. The new supply voltage and frequency levels of each core are finally set by the VRCM, which may actually choose different values than those recommended by the PM. Details will be discussed in the following subsections.

B. Reactive VRCon

The power saving achieved by employing DVFS strongly depends on the frequency of the decision making process, or equivalently, the duration of decision period (T_{DVFS}). If T_{DVFS} is small, the output of the VR and PLL will change more frequently, which results in better responsiveness to load changes but also higher energy loss and delay penalty due to overhead of DVFS transitions. T_{DVFS} should thus be considered a design variable to be set by the PM, which needs to be (much) longer than the voltage scaling time of the VR [24]. On the other hand, by turning on/off the network switches, the time to reconfigure the VR-to-core network (T_{NS}) is only limited by the transient response of the VR, which is in general much shorter than the voltage scaling time ($T_{NS} < T_{DVFS}$). Consequently, we treat the DVFS setting and network reconfiguration as the global and local power managements of VRCon, respectively. T_{DVFS} and T_{NS} are the required minimum global and local decision epoch lengths, respectively.

For its local power management function, the reactive VRCon applies only to cores operating at the same supply voltage level. As shown in Fig. 5, the blue box shows the cases when the reactive VRCon can be applied. The VRCM in this case performs only the network switch control to minimize the total energy consumption (that is, it will not change the voltage and frequency decisions of the PM). This total energy consumption is the summation of energy losses of the active VRs (including network switches) and the energy consumptions of the cores during the time period T_{DVFS}. We define T_l as the time period of l^{th} local management satisfying $T_l \geq T_{NS},\ for\ \forall l$, and $\sum_{l=1}^{L} T_l \leq T_{DVFS}$. Now then, the total energy consumption in T_{DVFS} can be expressed as:

$$E_{T_{DVFS}} = \sum_{i=1}^{N} E_{core,i} + \sum_{l=1}^{L}\left(\sum_{i=1}^{N} E_{NS,i,T_l} + \sum_{j=1}^{N} E_{VR,j,T_l}\right) \quad (4)$$

where minimizing the second term in (4) is the objective of the reactive VRCon. In the equation, N is the total number of cores. The energy consumption of the i^{th} core is given by $E_{core,i} = \int I_{core,i}(t) V_{core,i} dt$, where $I_{core,i}(t)$ is the input current of the i^{th} core, and $V_{core,i}$ is the input voltage of the i^{th} core. $I_{core,i}(t)$ is a function of time, but $V_{core,i}$ is constant in the period of T_{DVFS}. We define the energy loss of the turned-on network switch connected to the i^{th} core during time period T_l as E_{NS,i,T_l}. The energy loss of the j^{th} VR during time period T_l is defined as E_{VR,j,T_l}. For the local power management in an arbitrary time period, we use $E_{NS,i}$ and $E_{VR,j}$ to represent the general forms of E_{NS,i,T_l} and E_{VR,j,T_l}, respectively.

If identical power MOSFETs are used for the network switches, the power loss of the power MOSFET $P_{NS,i}$ may be expressed as [25]:

$$P_{NS,i}(t) = \frac{I_{on,i} + I_{off,i}}{I_g} V_D Q_g + \frac{1}{2} C_{OSS} V_D^2 + I_{core,i}^2\ R_{NS}, \quad (5)$$

where the first term is the switching loss during the turn-on and turn-off times; the second term is the switching loss from output capacitance of the power MOSFET; and the third term is the conduction power loss. I_{on}, I_{off} are the load current at the turn-on and turn-off times, I_g is the gate drive current; V_D is bus voltage; Q_g is the gate charge, which is generally provided in power MOSFET datasheets, and C_{OSS} is the output capacitance of the power MOSFET given by the gate-to-drain capacitance plus the drain-to-source capacitance of the switch. R_{NS} is the on-state resistance of the power MOSFET. From (5), we can derive $E_{NS,i}$.

To obtain $E_{VR,j}$, we could use the VR power loss model in [16], [15], or circuit simulations with the target VR module. Either method requires the load voltage and current values. The output voltage of a turned-on VR is set to be the supply voltage level of any core connected to the VR. On the other hand, the output current of the VR is set to be the sum of the load currents of the connected cores. Note that if the local power management aims to consolidate some cores to one VR, the maximum load current should not be greater than the maximum current rating of the VR. The red box in Fig. 5 shows the cases when the reactive VRCon can not be applied, because of the overrated combined load current.

Owing to the limited number of cores in each group of the connection networks, it becomes manageable to find the cores to be combined to minimize the energy consumption of both VRs and network switches in a group. To achieve this goal, VRCM first sorts the cores in each group that have the same voltage levels and a lower amount of input current than the maximum driving capability of a VR. Then, based on the current levels, VRCM finds the two cores, by merging which the VR energy saving is maximized. After consolidation of those two cores, VRCM keeps repeating this procedure until there is no core available, or the VR energy saving from the consolidation of the remaining cores is less than the power loss of the network switch transition.

C. Proactive VRCon

For its global power management function, the proactive VRCon exploits DVFS technique to perform frequency (and its corresponding voltage level) scaling taking into account energy consumptions of both cores and VRs, in the decision period, T_{DVFS}. In our proposed method, there exists a trade-off between the energy saving by DVFS (which is initially determined by the PM), and reduced energy loss by adaptively turning off the VRs and using fewer number of VRs at higher conversion efficiencies. If the VRCM finds out that the latter option is more desirable, the VRCM will not decrease the frequency/voltage levels of some cores to the minimum possible level; Instead, it will adjust the frequency/voltage levels of the cores to increase the opportunities for applying the VRCon procedure.

Compared to the reactive VRCon, the objective here is to find the frequency/voltage level of each core during T_{DVFS} to minimize the total energy consumption, which can be formulated as:

$$min\left(\sum_{t=1}^{T} E_{T_{DVFS,t}}(V_{core,1}, V_{core,2}, .., V_{core,N})\right), \quad (6)$$

where $E_{T_{DVFS,t}}$ denotes the total energy consumption during the t^{th} time period of T_{DVFS}, which is formulated in (4). $T_{DVFS,T}$ indicates that all the task processings are finished in this period. Given that $V_{core,i}$ in the period T_{DVFS} affects the results of the reactive VRCon, $E_{core,i}$, E_{NS,i,T_l} and E_{VR,j,T_l} in $E_{T_{DVFS,t}}$ are functions of $V_{core,i}$.

Because of (i) changing $V_{core,\forall i}$ in time period $T_{DVFS,t}$ affects the VRCon results in period $T_{DVFS,t+1}$, and (ii) the locking and synchronization issues of the multi-thread applications in multi-core processors, solving (6) is difficult. Therefore, by exploiting the initial DVFS schedule of the PM, we first divide the overall problem into sub-problems,

Algorithm 1 To find a set of the new voltage levels based on the proactive VRCon, under the homogeneous PDN

```
Initialization
define $S$: a set of the consolidated cores to a single VR ▷ $S$ is a subset solution for (7).
define $C = \{(I_1,V_1),(I_2,V_2),...,(I_M,V_M)\}$ ▷ $M$ is the number of cores in a group of connection network

function $Find\_Max\_Saving$ ($C$) ▷ Find two cores that achieves the maximum power saving by the consolidation.
    Find $i$ and $j$ such that
    $i \neq j,\ i,j \leq K,$ ▷ $K$ is the number of elements in $C$
    $V = max(V_i,V_j)$ ▷ Max. voltage level is chosen.
    $I = I_{i,new}+I_{j,new} \leq I_{out(max)}$ ▷ $I_{i,new}$ is the new current value of $i^{th}$ core indued by changing the voltage level of the core. If the voltage has not been changed, $I_{i,new} = I_i$.
    $max\begin{pmatrix} P_{loss}(I_i,V_i)+P_{loss}(I_j,V_j)-P_{loss}(I,V)+I_iV_i+I_jV_j \\ -(I_i+I_j)V+P_{NS}(I_i)+P_{NS}(I_j)-P_{NS}(I) \end{pmatrix}$
        ▷ Calculate the power saving. $P_{loss}$ and $P_{NS}$ are from (1) and (5), respectively.
    if $i$ and $j$ exist and the maximum power saving $> 0$ then
        update S, and return $\{(I_i+I_j),c_l : c_l \in C, l \neq i,j\}$ ▷ Now we treat these two cores as one equivalent core.
    else Return $\{\}$

function $VRCon\_pro\_I$ (C) ▷ Main function
    while $C \neq \{\}$ do
        $U = \{c|c \in C, I \in c \leq I_{out(max)}\}$
        Map $u \in U$ to $s \in S$ ▷ match the re-arranged $u$ to $s$
        $C = Find\_Max\_Saving\_I$ ($C$) ▷ A new set C is updated.
    return S
```

each of which only concerns how to modify the initial DVFS schedule to optimize the energy saving results of the reactive VRCon in a given period, T_{DVFS}. In order to guarantee that the performance (i.e., total execution time of applications) is not degraded by the modification of DVFS schedule, we impose the constraint that the VRCM can only keep the same or increase (but not decrease) the frequency/voltage level of each core from the original DVFS level suggested by the PM. Now, we transfer the problem in (6) to a problem to find the energy-efficient network configuration and voltage level of cores that minimize the total power consumption while maintaining the performance of the system. If we define the network configuration so that S_n denotes a set of the consolidated cores to the n^{th} VR, we can formally describe the problem as follows:

$$\begin{aligned} &\textbf{Find } N \text{ sets } S_1, S_2, ..., S_N \\ &\textbf{to minimize } E_{T_{DVFS},t}(V_1,V_2,..,V_N) \\ &\textbf{Subject to } \; V_{S_n} = \max_{m \in S_n}(V_m), \text{ and } I_{S_n} = \sum_{m \in S_n} I_{m,new} \leq I_{out(max)} \end{aligned} \tag{7}$$

where V_m, $1 \leq m \leq N$, is the voltage level suggested by the PM of the m^{th} core; V_{S_n} is the maximum voltage levels of cores consolidated to the n^{th} VR (those n^{th} set), $I_{m,new}$ is the new current value of the m^{th} core under V_{S_n}; I_{S_n} is the summation of I_m's, $m \in S_n$. If the VRCM finds a solution to the above problem, it will override the DVFS level recommended by the PM with the new voltage level.

From the assumption that tasks during time period T_{DVFS} have already been assigned to the cores according to the PM's recommendation, we focus only on the DVFS decisions of the VRCM without any task migration. Consequently, (7) can be divided into a set of subproblems, each of which is to find DVFS levels only for the cores belonging to the same group. Furthermore, the number of cores in any group is constrained by the maximum load current $I_{out(max)}$ that a single VR can drive. Therefore, it is tractable to search all possible DVFS levels of the cores in the same group when only voltage increases are possible. We have implemented a clustering-based heuristic solution as shown in Algorithm 1. We first sift through the cores in a group driving a small amount of current so that they can be combined with others. In order to respond to the dynamically changing current, we determine the amount of current of each core by the average current during the (previous) decision period, T_{DVFS} (i.e., in the proactive VRCon, we first determine the voltage levels of the cores and the network configuration. Later, during the current decision period, the reactive VRCon changes the network configuration according to the dynamically changing current of cores in real time.) Next we perform the function, $Find\_Max\_Saving$, in Algorithm 1 to find the two cores and their voltage level that can achieve the maximum power saving, if they are merged with the same voltage level. We then treat these two cores as one equivalent core. The procedure is repeated until no energy saving can be achieved by VR consolidation, in the function of $VRcon\_pro\_I$.

Notice that if the VRCM gets involved in the task allocation to the cores, and the target platform has a large number of cores, then solving (7) may require more sophisticated combinatorial optimization approach to find the best core to VR matches. This is, however, outside the scope of the present paper. Instead, interesting readers may refer [26], [27] that had considered the issues in the hardware-software cosynthesis and codesign.

D. Design considerations

Compared to the conventional per-core DVFS platforms where each core has a single dedicated VR, our proposed network switches will incur additional energy losses. Precisely, the switching energy loss of the i^{th} network switch, $E_{NS,switching,i}$, the first and second term in (5) have a direct effect on the time period of the reactive VRCon, T_{NS}. In general, the lower bound of T_{NS} can be determined by:

$$max[\delta_i \cdot Delay_{NS,i}, \text{ for } 1 \leq l \leq N] \leq T_{NS}, \tag{8}$$

where δ_i is the transition factor, and $Delay_{NS,i}$ is the delay of the network switch that powers the i^{th} core. Interesting readers may refer [28] that describes the detailed way to calculate $Delay_{NS}$ by using the power MOSFET parameters in datasheet. If the i^{th} core changes its network switch, $\delta_i = 1$, otherwise, $\delta_i = 0$. Then the set of δ_i is derived from:

$$\sum_{i=1}^{N} \delta_i E_{NS,switching,i} \leq Gain_{VRCon}(T_{NS}), \tag{9}$$

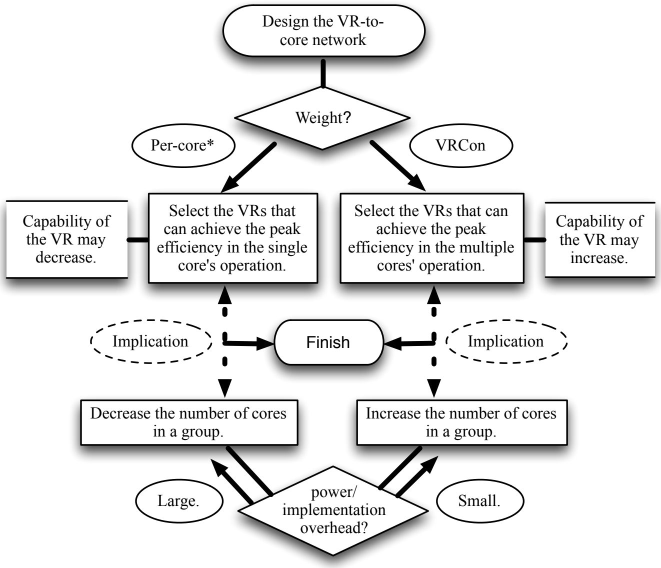

Design the VR-to-core network
Weight?
Per-core*
VRCon
Select the VRs that can achieve the peak efficiency in the single core's operation.
Select the VRs that can achieve the peak efficiency in the multiple cores' operation.
Capability of the VR may decrease.
Capability of the VR may increase.
Implication
Finish
Implication
Decrease the number of cores in a group.
Increase the number of cores in a group.
Large.
power/ implementation overhead?
Small.

Fig. 6. Design flows to determine the VRs and the number of network switches in the proposed platform. Per-core* in the figure means that a designer puts more weight on the energy saving of the VR by setting it to achieve the best efficiency in the normal operation condition of each core.

where $Gain_{VRCon}(T_{NS})$ is the total energy that can be saved from the reactive VRCon during time period T_{NS}.

Regarding to selecting the network switches, the following should be considered. In (5), $E_{NS,switching,i}$ is proportional to the charge of the switches, whereas the conduction energy loss, $E_{NS,conduction,i}$, is affected by R_{NS}. Therefore, if the switch transition occurs frequently in a short time, selecting a power MOSFET that offers the smaller charge values may be preferrable. In contrast, if T_{NS} is long enough, $E_{NS,conduction,i}$ would become the dominant source of $E_{NS,i}$. Then designers would better focus on choosing the smaller R_{NS}. Of course, the area overhead due to the network switches should be carefully determined at the design time.

Selecting the VRs is another important concern in the proposed platform. The VR has limited capability to provide a large amount of load current, as mentioned above in Section II. Typically, the VRs that have the higher load current capabilities are equipped with power MOSFETs that offer the smaller resistance but relatively higher charges. Therefore, these VRs perform their peak conversion efficiencies in the higher load current region than the peak conversion efficiency region of the VRs that have the lower load current capabilities. If the VRs with larger capabilities are selected (i.e., these VRs will achieve peak conversion efficiency in the higher load current region than the normal load current of each core), the potential power saving from VRCon could be much higher than the case when each VR is optimally chosen to power a single core (in this case the VR achieves peak efficiency at the normal operation condition of a single core). Nevertheless, we should also consider the later case that accords with the original setup of the VRs for the per-core DVFS: each VR is dedicated to power a single core with the best VR efficiency. In this paper, for the fair comparison between the results from applying VRCon and not, we use the same platform for both cases that adopts one type of the VRs, each of which is set to achieve the high efficiency in the normal operation region of the core, or each of which has the high load current capability. We calls this setup as a homogeneous PDN. Later in Section IV, we will also discuss a heterogeneous PDN that is composed of two different types of VRs, one for VRCon and the other for the operation of the core.

Meanwhile, the capability of the VR also affects the number of cores in a group. In other words, due to the limited capability of the VR, the number of cores that can be connected to one VR should be limited. Therefore, designing the VR-to-core network to support all the connections between all the VRs and cores is redundant. In addition, the output voltage fluctuation (a.k.a., voltage droop [29]) problem should be taken into consideration. Because a rapid and large change of the load current of a VR can cause a critical output voltage swing of the VR, more than a certain number of cores should not be connected to one VR at once. We thus proposed the network grouping, where only the VRs and cores grouped in the same subnetwork can be connected. This is also described in Fig. 4. Furthermore, owing to the limited numbers of connections between the network switches and cores, this grouping can mitigate the scalability issue that the power/implementation overhead from the network switches becomes more significant as the platform is equipped with more cores.

Finally, we present the design flows in Fig. 6 to select the VRs and determine the number of cores in a group. A designer first selects the VRs after deciding where to put more weight on, between the benefits from the VRs optimized for the normal operation condition of each core, and advantage of the VRCon by using the VRs offering the high capabilities. If the designer chooses the first, then the number of cores in a group may be smaller than that from the case when the designer chose the later. According to the required design specification that allows the power/implementation overhead of the network switches, the designer may need to retrace the flows, in such a way that the designer increases/decreases the number of cores in a group, and even select the VRs again.

IV. Heterogeneous PDN

In the previous section, we have discussed the relationship between the effectiveness of VRCon and the current driving capability of a single VR in the homogeneous PDN. When the VR is selected to achieve its highest efficiency in the normal operation region of each core, the current driving capability of switches in the VR may be relatively small (we call this VR a little VR), and in this case we can achieve limited power saving from VRCon. On the other hand, selecting the VRs with a higher capability (we call this VR a big VR) can increase the power savings from the VRCon, while losing the benefits from selecting little VRs when VRCon is not applied. Therefore, selecting VRs in a target homogenous PDN requires accurate estimation of how often the VRCon will be applied and how much energy saving will result from the VRCon. However, these information may be difficult to obtain at the design stage. Then an inaccurate estimation can lead to mismatched VRs, thereby losing both benefits of little VRs and big VRs.

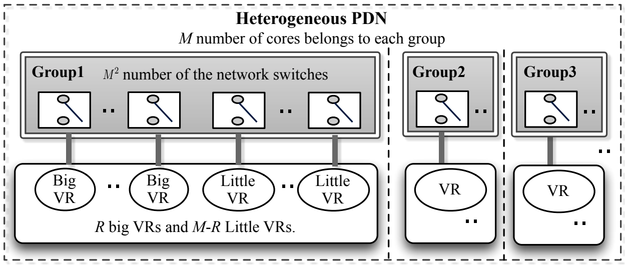

Heterogeneous PDN
M number of cores belongs to each group
Group1
M2 number of the network switches
Group2
Group3
Big VR
Big VR
Little VR
Little VR
VR
VR
R big VRs and M-R Little VRs.

Fig. 7. A part of the proposed platform with the heterogeneous VRs: each group has the R big VRs and M-R little VRs

To overcome the drawback of the homogeneous PDN, we present the heterogeneous PDN comprised of big and little VRs. The heterogeneous PDN represents a desirable tradeoff between two extremes of selecting only big VRs or only little VRs. In this paper, we consider the two (big and little) types of VRs instead of various types of VRs due to the following reasons: i) Using only two types of VRs reduces the control complexity. Applying VRCon to the heterogenous PDN must solve a problem to find optimal connections between cores and the VRs that are not the little VR. If there exist many types of such big VRs, the complexity of the problem may significantly increase, which may result in a heavy computation complexity of the VRCM. ii) Because the number of cores and VRs in a group is limited (due to the power/implementation overhead), the possible range of the load current of all cores in a group is limited. If the current range is not too wide, using two types of VRs may be enough to improve the efficacy of VRCon in the heterogenous PDN.

We first explore a heterogeneous PDN design that has the same number of little VRs as in the homogeneous PDN, but is equipped with extra big VRs in each group. This design enjoys both benefits (of little VRs and big VRs), in that the little VR achieves high efficiency by powering a single or a few number of cores, and the big VR takes responsibility for the consolidation of a large number of cores. However, adding extra big VRs may not yield commensurate benefits that justify the area/implementation overheads. For instance, if there are M cores in a group, adding R extra big VRs in a group requires an additional $M \cdot R$ network switches and additional wire connections to the VR. The overheads will be exacerbated as the number of cores embedded in the platform increases. More precisely, if the big VR consists of the LTC3816 converter (area: $35mm^2$ and cost: \$4.8 [11]) with two Si4442DY power MOSFETs (each, $27mm^2$ and \$3.25 [30]), one 7447709100 inductor (69mm2, \$3.1) and three EEE-1EA100WR capacitors, (each, $12mm^2$, \$0.5), one big VR at least occupies $194mm^2$, and requires \$15.9 – the device prices are taken from [31]. Moreover, if there exists 8 cores in a group, adding one big VR needs 8 more network switches, which may induces $216mm^2$ area overhead and \$26 additional cost.

A. Proposed design of the heterogeneous PDN

Adding extra big VRs to the existing PDN with little VRs can not avoid the scalability issue. Therefore, instead of adding redundant devices, we propose the heterogeneous PDN that replaces R little VRs by the same number of big VRs in each group. Consequently, the total number of VRs assigned to one group is the same with that in the homogeneous PDN design. Fig. 7 illustrates the proposed design of heterogeneous PDN, as a part of the proposed platform in Fig. 4.

Algorithm 2 To determine the number of the big VRs in a group and the power MOSFETs inside the big VRs

```
Initialization
define $Gain(R,Cap)$        ▷ $Gain(R,Cap)$ is the energy savings of both cores and VRs for the given load condition profile, when the $R$ number of the big VRs in a group are replaced. The capability $Cap$ of the switches are attached to the big VR.
$g = Gain(0,Cap_{little})$ ▷ $R = 0$ implies that no big VR is required.

function $Find\_R\_W_{big}$ (Load condition profile)
    for $1 \leq m \leq M$ do            ▷ To find i) the number of big VRs.
        $Cap = Cap_{little} + \Delta Cap$
        while $g < Gain(m,Cap)$ do   ▷ ii) the cap. of the big VR
            $g = Gain(m,Cap)$, $Cap_{big} = Cap$, and $R = m$
            $Cap = +\Delta Cap$         ▷ $\Delta Cap$ is the min. cap. increase.
        if $Cap = Cap_{little} + \Delta Cap$ then        ▷ this is the case that increasing $R$ can not bring the better power saving.
            break
    return ($R$, $Cap_{big}$)
```

In order to determine how many little VRs should be replaced by big VRs, and how to select the powerFETs for the big VRs, we first need to estimate the load conditions of all the cores in a group. Recall that the homogeneous PDN design has a risk that inaccurate estimation of the load conditions may cause mismatch between VRs and actual load conditions, which can cause a significant amount of VR power losses. In contrast, using both big and little VRs simultaneously can mitigate the risk from inaccurate estimation. Hence, we can use the load profiles collected by running various benchmarks on the target platforms to estimate the load conditions.

Let R denote the number of big VRs in a group, and Cap_{big} and Cap_{little} denote the current driving capability of the switches inside the big VR and the little VR, respectively. The objective here is to find such R and Cap_{big} values to maximize the power gain, which is the power saving by applying VRCon subtracted by the power loss from VR mismatches. We present a heuristic solution that starts from replacing one little VR by a big VR in a group. Then we keep increasing Cap_{big} from Cap_{little} and testing the big VR equipped with the corresponding power MOSFETs, until the increased Cap_{big} no longer improves the power gain (cf. the while-loop in the algorithm 2). Next, we increase R to two, followed by increasing Cap_{big} of the two big VRs to search whether this increase results in higher power gain than the value obtained previously with one big VR (cf. the for-loop in the algorithm 2). We repeat these procedures until we can not achieve higher power gain. Algorithm 2 explains the proposed procedure in detail.

B. VRCon for the heterogeneous PDN

It is an NP-hard problem to apply VRCon to the proposed heterogenous PDN to find the best connections between VRs and cores to save the maximum amount of energy. To prove the NP hardness of the problem, we reduce the problem to only maximize the energy savings from VRCon, but ignore the energy consumption induced by VR-to-core mismatches. Then the problem is now transformed to a *generalized assignment problem*, which can be formulated as follows:

Given that there are M (heterogeneous) VRs and M cores in a group, each VR has a limited driving capability of its total load current, and each core has a required load current level. Any VR can be assigned to power a subset of cores, as long as the sum of the load currents of assigned cores does not exceed the limit. Depending on the VR-to-core assignment, the profit (i.e., power saving) of each VR varies. The objective is to find an assignment in which the total profit is maximized. If this problem is further simplified so that the profit is only a function of load current, but not affected by the types of VRs, the problem becomes a sort of *multiple knapsack problem* that is a well known NP-complete problem in combinatorial optimization.

We propose heuristic algorithms to apply the reactive and proactive VRCons to the heterogeneous PDN. We first attempt to maximize the utilization of the big VRs in the proposed algorithms. In general, utilizing bigger VRs can give rise to turning off more little VRs and mitigating the energy loss incurred by the mismatches between big VRs and their assigned cores. This approach can also significantly reduce the computational overhead because we do not need to enumerate all the possible connections between all the cores and VRs. At the beginning of this step, we set one big VR as the target VR, and estimate the benefit of each core if the core is connected to the target VR. We define *profit* for each core as the power saving that can be acquired from assigning the core to the big VR and turning off the little VR. Then the *profit* of each i^{th} core is calculated as follows:

$$P_{loss,little}(I_i,V_i) - P_{NS,i} - I_i(V_{base} - V_i) \tag{10}$$

where I_i and V_i are the load current and voltage levels of the i^{th} core, respectively, $P_{loss,little}$ is the power loss of the little VR in (1), and $P_{NS,i}$ is the power loss during the network switch transition. Notice that, to calculate $P_{loss,little}$, we suppose that the core is currently connected to a dedicated little VR, regardless of what type of VR the core is actually connected to. This is reasonable because any core should be connected to a little VR if it is not connected to a big one. On the other hand, the current connection between the core and VR is taken into consideration during the calculation of $P_{NS,i}$. If the core is connected to a big VR, $P_{NS,i}$ is zero, otherwise the transition incurs power dissipation $P_{NS,i}$. The third term is the estimate of power loss from the potential voltage level change. V_{base} is thus equal to V_i when the reactive VRCon is applied. For the case of the proactive VRCon, we set V_{base} to the most common level (or the medium level) among all the voltage levels of the cores.

Then we perform a procedure to select the cores that are connected to the big VR. More precisely, the problem here is to find a subset of cores, such that the sum of their *profits* is maximized and the sum of their current values is less than or equal to the limit of a big VR. This problem is similar to the well-known *Knapsack problem*, so that we can exploit a *dynamic programming* to solve the problem in pseudo-polynomial time.

After assigning the cores to the target big VR, we repeat above procedures for the other big VR, until all the big VRs are investigated or there exists no available core. If there remains cores that are not connected to the big VRs, we now exploit the VRCon algorithms that we have presented for the homogenous PDN in Section III. To assign the rest of the cores to the little VRs, for example, $VRCon\_pro\_I$ in Algorithm 1 is used here again. Similarly, the reactive VRCon for the heterogenous PDN in this step is the same as the reactive VRCon for the homogeneous PDN that we have discussed in Section III-B.

V. Experimental work

A. Experimental setup

1) per-core DVFS, multi-core processor setup: Unlike the conventional platform, the VRCM in our proposed platform performs DVFS referred to the PM's initial recommendation. We thus treat the PM's DVFS recommendation as given *a priori* in this paper, exploit an offline DVFS approach as an intermediate step for the overall aim. Similar to [4], we adopt an ILP based algorithm.

Finding the optimal frequency/voltage level of each core to minimize the energy consumption under a certain performance penalty, β, may be formulated to:

$$min\left(\sum_{r}^{R}\sum_{s}^{S}P_{r,s}x_{r,s}\right)$$

$$s.t.\ \sum_{r}^{R}\sum_{s}^{S}D_{r,s}x_{r,s} < \beta\ ,and\ \sum_{r}^{R}\sum_{s}^{S}x_{r,s} = R \tag{11}$$

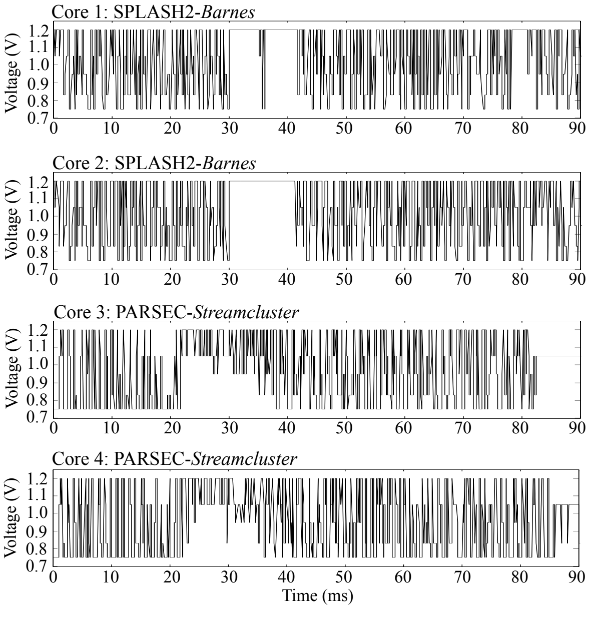


Fig. 8. A part of the per-core DVFS results of *Barnes* and *Streamcluster* from the Sniper simulation with 4-core setup.

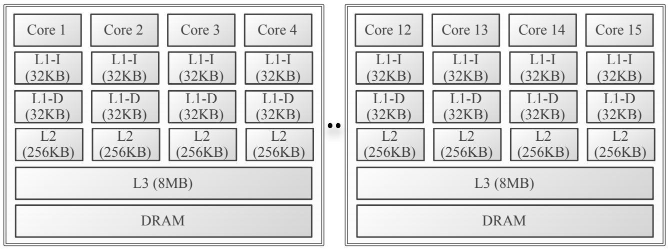


Fig. 9. Topology of 16 cores (four 4-core processors) in Sniper simulation.

TABLE I
DVFS FREQUENCY AND VOLTAGE LEVELS.

GHz, V	2.66, 1.2	2.33, 1.05	2.13, 0.95	1.87, 0.83	1.66, 0.75

where R is the total number of intervals, and S is the set of the five frequency/voltage levels described in Table I. $P_{r,s}$ is the power consumption when running at s^{th} frequency/voltage level for r^{th} interval. By following the same notation to $P_{r,s}$, $D_{r,s}$ denotes the incurred delay under the frequency/voltage condition. To obtain $P_{r,s}$, $D_{r,s}$, we first performed detailed multi-core simulations for various benchmarks under the five frequency/voltage levels. From the simulation set by the highest frequency/voltage level, the intervals and the default instructions count for each interval were acquired. Based on the default instruction counts, $P_{r,s}$, $D_{r,s}$ were then derived. Finally, IBM CPLEX was used to solve (11). Fig. 8 shows an example of the offline DVFS results from $\beta = 15\%$, for two applications in the 4-core simulator setup.

We performed the multi-core processor simulations in the Sniper simulator. The platform configurations were set based on Intel Xeon Nehalem architecture, the topology is shown in Fig. 9. We modified the codes related to the McPAT module in the Sniper to collect the power and timing data from per-core DVFS. The multi-threaded applications from the PARSEC and SPLASH2 benchmarks were used in the simulation.

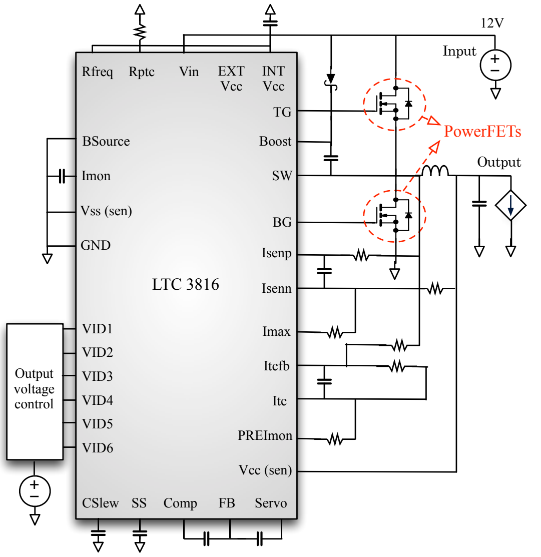


Fig. 10. VR schematic used in the spice simulation.

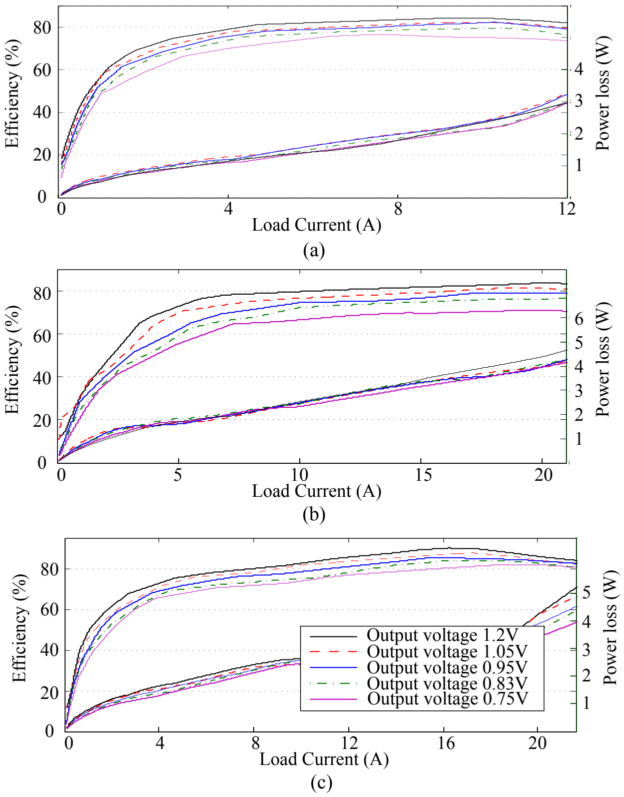


Fig. 11. Efficiency and Power loss vs. Load current for LTC3816 with (a) Si4840DY, (b) Si4838DY and (c) Si4442DY.

2) *Homogeneous PDN setup:* We selected the programmable VR from Linear Technology, LTC3816 [11], which satisfies the Intel VR-design guideline (VRD 11.1 [32]), and can power each core in our processor setup with the five output voltage levels. Next, we selected Si4840DY for the power MOSFETs, which is a N-channel trench power MOSFET from Vishay Siliconix [33]. The on-state resistance and charge and the maximum continuous drain current of Si4840DY are $9m\Omega$, $19nC$ and $12.4A$, respectively. We then performed LTspice simulation based on the circuit diagram shown in Fig. 10. Fig. 11 (a) shows the resulted VR efficiencies according to the various load current under the five output voltage levels. We set the input voltage level to 12V followed by the VRD 11.1. Given that the load current profiles of a single core gathered from the various benchmark simulations in the Sniper simulator resulted that the typical load current ranged from $4A$ to $10A$, and the maximum current was less than $12.4A$, the simulation results show that LTC3816 with Si4840DY is tailored to the dedicated VR for the single core in our multicore setup.

We performed additional homogenous PDN simulations with a different VR setup, in order to investigate the effect from the VR mismatch. As aforementioned, the VR mismatch occurs if we select the power MOSFETs to let the VR have the larger current driving capability, but the induced best efficiency region of the VR may be higher than the load current region from the normal operation of a single core. In reality, when selecting VRs, designers put the high priority on the capability of the VR so that the VRs can drive the maximum (possible) load current of a target core [11] (i.e., however, the load current from the normal operation of the core can be much less than the maximum load current). Furthermore, reference [15] showed that real (smartphone) devices can be equipped with some VRs that are set to achieve their best efficiencies in the vicinity of the maximum load current value. In light of these, we selected Si4838DY [34], which has the on-state resistance R_{DS} and charge Qg and the maximum continuous drain current I_D, $3m\Omega$, $40nC$ and $25A$, respectively, to be incorporated at LTC3816. The resulted efficiency of LTC3816 with Si4838DY from the LTspice simulations is shown at Fig. 11 (b). This figure shows that the efficiency of LTC3816 with Si4838DY is less than LTC3816 with Si4840DY in the region less than 12A, but can drive the higher load current.

For the network switch, we select SiR800DP that has the lowest resistance ($2.3m\Omega$) among the power MOSFETs from Vishay Siliconix, which is also available in LTspice simulation. SiR800DP has $40nC$ on-state charge and occupies $32mm^2$ area. By taking account of the load current driving capability of the VR and power/area overhead of the network switches, we set the number of VRs and cores in one group of the VR-to-core networks to four.

3) Heterogenous PDN setup: As we discussed in Section IV, in order to mitigate the overheads of the big VRs in the heterogenous PDN, we chose to replace R little VRs by the same number of big VRs. And, we limit one network group to support only connections between four cores and four VRs.

We used LTC3816 as the big VR, which was also used in the previous homogeneous PDN, but we changed the power MOSFET inside LTC3816 so that the big VR has a higher current driving capability Cap_{big}. To determine such a power MOSFET, we first set the baseline: a homogenous PDN that employs little VRs with Si4840DY power MOSFET (Cap_{little}=12.4A). We used a testbench to perform Algorithm 2, which was one of the DVFS results in Section V-A1. Precisely, we set one network group to have four cores, and ran Barnes in two of the cores and FMM in the other two cores. The performance penalty β of the testbench was 15%. Next, we investigated the Vishay power MOSFETs, such as Si4114DY, Si7106DN, Si4442DY and Si4838DY, as introduced in Table II. We performed Algorithm 2 to find the power MOSFET equipped in the big VR and the number of big VRs that could achieve the highest Gain. For readers' better understanding, we define the total VR energy loss reduction $G_{VR}(\%)$ and the total energy saving in the platform $G_{total}(\%)$. Table II shows that replacing a small VR by a big VR that includes Si4442DY as the power MOSFET [30] results in the highest improvement.

TABLE II

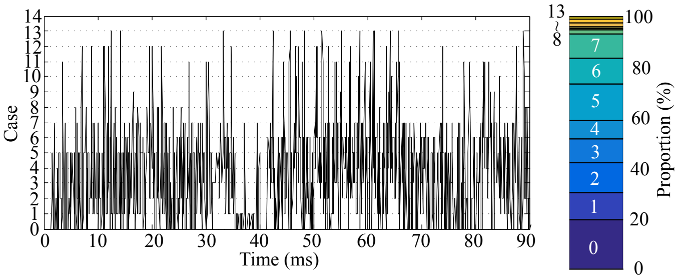
DESIGN PROCEDURE TO BUILD THE HETEROGENEOUS PDN, FOLLOWING BY ALGORITHM 2: THE BASELINE (HOMOGENOUS) PDN WITH SI4840DY (Cap_{little}=12.4A) ARCHIVES $G_{VR} = 22.56\%$, AND $G_{total} = 5.56\%$. DETAILS ARE DESCRIBED IN SECTION V-A3.

Big VR	Cap_{big}	R	$G_{VR}(\%)$	$G_{total}(\%)$
Si4114DY	15.2A	1	23.90	5.89
Si7106DN	19.5A	1	24.05	5.93
Si4442DY	22A	1	24.66	6.09
Si4838DY	25A	1	16.06	3.96
Si4114DY	15.2A	2	24.18	5.96

Finally, we selected the power MOSFET Si4442 for the big VR in Table II. Si4442DY has on-state resistance R_{DS} and charge Q_g of $5m\Omega$ and $36nC$, respectively, whereas its maximum drain current I_D is $22A$. As aforementioned, due to the smaller resistance but higher on-state charge and I_D of Si4442DY than those of Si4840DY, this big VR is less efficient than the little VR if the current is low, but achieves high efficiency in the high current region. In other word, this big VR can drive the higher load current with high efficiency than the little VR. Fig. 11 (c) shows the efficiency of the big VR, where its driving current capability is 22A. We determined the number of big VRs to one in one group. Indeed, the improvement by exploiting the big VR in Table II is not so distinguishable. This is because the given load current profiles from the benchmarks were well matched to the homogeneous PDN with the little VRs. However, if the cores run into some different load current conditions that were not captured by the used benchmarks, the need to use the big VRs should be enlarged. For instance, one case that the four cores have 1A, 1A, 1A and 12A results that one big VR can power all the cores with high efficiency, but the homogenous PDN has to use two little VRs, and one of them has the load current just 3A that corresponds to very low efficiency (cf. Fig. 11 (a)). We will discuss this later in Section V-B3.

B. Simulation results

1) Homogeneous PDN composed of the VRs with Si4840DY (simply called well-matched PDN): Following Section III-B and III-C, we performed the reactive and proactive VRCon (cf. Algorithm 1) in the homogeneous PDN. Fig. 12 shows the proactive VRCon result of the per-core DVFS example described in Fig. 8. In the figure, the voltage levels of some of the cores in certain decision epochs are changed from their initial levels for the VR consolidation, or some of the cores are consolidated without voltage level change. Fig. 12 also provides a histogram to show how often the



Case	Result*	Case	Result*	Case	Result*
0	(1, 2, 3, 4)	5	(1, 2), (3, 4)	10	(1, 4), (2), (3)
1	(1, 2, 3), (4)	6	(1, 3), (2, 4)	11	(2, 3), (1), (4)
2	(1, 2, 4), (3)	7	(1, 4), (2, 3)	12	(2, 4), (1), (3)
3	(2, 3, 4), (1)	8	(1, 2), (3), (4)	13	(3, 4), (1), (2)
4	(1, 3, 4), (2)	9	(1, 3), (2), (4)	14	(1), (2), (3), (4)

Tuples in Result* indicate the consolidated sets of cores

Fig. 12. VRCon result from Fig. 8.

TABLE III
SIMULATION RESULTS OF THE HOMOGENEOUS PDN (VRS WITH SI4840DY): APP.*, β, RE.*, PRO.*, G_{VR}(%) AND G_{total}(%) INDICATE THE APPLICATION, DVFS PERFORMANCE PENALTY, REACTIVE, PROACTIVE, VR ENERGY LOSS REDUCTION AND TOTAL ENERGY SAVING, RESPECTIVELY.

App.*	VRCon	$\beta = 5\%$		$\beta = 10\%$		$\beta = 15\%$		App.*	VRCon	$\beta = 5\%$		$\beta = 10\%$		$\beta = 15\%$	
		G_{VR}	G_{total}	G_{VR}	G_{total}	G_{VR}	G_{total}			G_{VR}	G_{total}	G_{VR}	G_{total}	G_{VR}	G_{total}
Streamcluster (I)	Re.*	24.68	6.23	19.01	4.88	16.19	4.16	*Swaption* (I)	Re.*	20.82	5.77	20.82	5.77	20.82	5.77
	Pro.*	28.81	7.28	23.19	5.95	20.21	5.19		Pro.*	24.34	6.75	24.34	6.75	24.34	6.75
Barnes (II)	Re.*	23.78	5.80	23.00	5.86	21.42	5.45	*FFT* (II)	Re.*	6.21	1.13	6.42	1.22	6.51	1.29
	Pro.*	32.21	7.86	31.30	7.98	29.50	7.51		Pro.*	6.40	1.16	6.59	1.25	6.70	1.33
Ocean (III)	Re.*	15.43	4.07	16.12	4.31	16.30	4.34	*Raytrace* (III)	Re.*	17.74	3.33	23.24	4.77	27.28	5.95
	Pro.*	19.11	5.04	19.74	5.28	19.77	5.26		Pro.*	18.09	3.40	22.96	4.71	27.52	6.01
Cholesky (III)	Re.*	12.84	3.17	15.34	4.04	15.39	4.13	*FMM* (III)	Re.*	10.02	2.23	11.63	2.75	11.34	2.68
	Pro.*	18.99	4.70	21.54	5.68	21.46	5.75		Pro.*	16.04	3.57	17.73	4.20	17.21	4.07

consolidation occurs. As aforementioned, by defining the total VR energy loss reduction as G_{VR} and the total energy saving in the platform as G_{total}, from the baseline VR and platform energy consumption (note that these baselines are resulted from the initial DVFS setup derived from (11)), the result in Fig. 12 achieves $G_{VR} = 15.45\%$, and $G_{total} = 4.02\%$. If only the reactive VRCon were applied, $G_{VR} = 12.44\%$, and $G_{total} = 3.24\%$.

We performed simulations on various applications under the different simulator setups (different number of cores) and different initial DVFS recommendations (derived from three different performance penalties). Table III shows the results. The number in the application name indicates the simulation setups: (I), (II) and (III) are for the 16-core, 8-cores and 4-cores setups, respectively.

While *Streamcluster*, *Barnes* and *Raytrace* resulted more than 25% G_{VR}, others except *FFT* achieves around 20% G_{VR}. Especially, *Barnes* improved 32% VR energy loss reduction which achieved 8% total energy savings. The reason why the gains of *FFT* were small may be because the load current values of each core from *FFT* are so high that (i) the sum of the load current values may be over the capability of the single VR or (ii) the efficiency corresponding to each load current value is already high, so the increased efficiency from the consolidation may not be distinguishable. In addition, *Swaptions*, as an example of memory-bound application, where no performance degradation was observed despite DVFS level drops, its initial DVFS recommendations for the three performance penalties are the same. That is why the VRCon results of *Swaption* for different β values show the same improvements in the table.

2) Homogeneous PDN composed of the VRs with Si4838DY (simply called mismatched PDN): We then performed simulations on the same applications in Table III, but exploiting the mismatched PDN. Table IV shows the improvement results from the case of each application that the DVFS performance penalty, β, is 15%. We defined $loss_{mis}$ to indicate how much (%) the total energy increased by changing the well-matched PDN to the mismatched PDN. The table shows that $loss_{mis}$ can be upto 11%. Note that the gains here were derived based on the total and VR energies from the mismatched PDN without the reconfigurable setup, not based on the energies from the well-matched PDN setup. Except the gains of *Swaption* and *Ocean* that become slightly reduced, the gains of all the applications, including $G_{VR} = 42\%$ from *Raytrace*, shows the increased results than corresponding results in Table III. This implies that the efficacy of the VRCon may become more powerful, as we discussed in Section III-D.

TABLE IV
SIMULATION RESULTS OF THE MISMATCHED HOMOGENOUS PDN. $loss_{mis}$ IS THE TOTAL ENERGY INCREASE (%) COMPARED TO THE TOTAL ENERGY WITH THE WELL-MATCHED HOMOGENOUS PDN.

Application ($\beta = 15\%$)	$loss_{mis}$ (%)	Reactive		Proactive	
		G_{VR}	G_{total}	G_{VR}	G_{total}
Streamcluster (I)	12.07	18.72	6.38	27.99	9.54
Swaption (I)	9.02	17.80	6.00	21.51	7.25
Barnes (II)	7.69	22.17	6.82	29.77	9.16
FFT (II)	7.91	7.98	2.05	8.45	2.17
Ocean (III)	10.04	16.52	5.50	18.89	6.29
Raytrace (III)	11.21	39.75	11.81	42.35	12.58
Cholesky (III)	10.21	19.06	6.41	24.82	8.34
FMM (III)	6.84	13.69	3.91	18.99	5.42

3) Heterogeneous PDN: We finally performed the heterogenous PDN simulations, following Section IV-B. We first explored the same applications used in the homogenous PDN simulations. For the fair comparison, the gains here were calculated based on the VR and total energies resulted from the well-matched PDN without the reconfigurable setup. Table V shows the resulted gains, that the results from the applications except *Streamcluster*, *Swaption* and *Ocean* become higher than the results from the simulations with well-matched PDN.

However, the applications in Table V may not encompass all the operating conditions of the cores, which may demonstrate the more superiority of the heterogenous PDN. In other words, as aforementioned in Section V-A3, there can be certain load current conditions of the cores, where the VRCon in the heterogenous PDN can achieves prominent power savings while the VRCon in the homogenous PDN can not. In order to capture such conditions, we manipulated three scenarios: (i) Scenario 1: one core kept 12A load current condition but the others kept loading 1A, (ii) Scenario 2: From the case of *Streamcluster* with $\beta = 15\%$, we added 10A to the load current condition of only one core, (iii) Scenario 3; the same setup to the Scenario 2, but we used *Radiosity*. The simulation

TABLE V
SIMULATION RESULTS OF THE HETEROGENEOUS PDN. G_{VR} AND G_{total} ARE THE GAINS FROM THE PROACTIVE VRCON.

Application	$\beta = 5\%$		$\beta = 10\%$		$\beta = 15\%$		Application	$\beta = 5\%$		$\beta = 10\%$		$\beta = 15\%$	
	G_{VR}	G_{total}	G_{VR}	G_{total}	G_{VR}	G_{total}		G_{VR}	G_{total}	G_{VR}	G_{total}	G_{VR}	G_{total}
Streamcluster (I)	16.95	4.28	12.09	3.10	9.45	2.43	Swaption (I)	14.74	4.09	14.74	4.09	14.74	4.09
Barnes (II)	36.46	8.89	33.21	8.47	30.34	7.73	*FFT* (II)	12.67	2.30	12.00	2.29	12.13	2.41
Ocean (III)	10.97	2.89	11.69	3.13	11.90	3.17	*Raytrace* (III)	19.72	3.72	21.31	4.38	28.66	6.25
Cholesky (III)	19.65	4.87	20.72	5.47	19.93	5.34	*FMM* (III)	19.94	4.44	21.88	5.19	20.41	4.83

TABLE VI
RESULTS OF THE BOTH HOMO- AND HETEROGENOUS PDN FROM THE THREE SCENARIOS. GAINS ARE FROM THE PROACTIVE VRCON.

Application	Homogenous PDN		Heterogenous PDN	
	G_{VR}	G_{total}	G_{VR}	G_{total}
Scenario 1	11.45	2.15	44.14	8.31
Scenario 2	6.45	1.24	18.89	3.64
Scenario 3	10.96	2.11	28.96	5.59

results are shown in Table VI. As seen, the VRCon gains from the heterogenous PDN show much higher than those from the homogenous PDN.

VI. CONCLUSIONS

This paper addressed the problem of power conversion efficiency in the multicore platform, where significant power is dissipated by the multiple VRs, and design limitations associated with the fixed VR-to-core network undermine the opportunity of power savings from the per-core DVFS technique. This paper proposed the VR consolidation methods with the configurable VR-to-core distribution network integrated in the proposed multicore platform design. The reactive VRCon was presented to configure the network to enhance the power conversion efficiency under the predetermined DVFS levels. The proactive VRCon was proposed to determine new DVFS levels for maximizing system-wide energy saving without performance degradation. We applied the proposed optimization methods to the PDN composed of homogeneous VRs, and demonstrated that the proposed method accomplish upto 32% VR energy loss reduction. Then we explored the limitation of the homogenous PDN, and proposed the heterogenous PDN that can increase the benefits of the optimization methods by incorporating VRs with a larger driving capability of load current. The simulation results based on the realistic experimental setups demonstrated that the proposed methods achieve upto 36% VR energy loss reduction and 9% total energy saving.

REFERENCES

[1] W. Lee, Y. Wang, and M. Pedram, "VRCon: Dynamic reconfiguration of voltage regulators in a multicore platform," *in Proc. Design Automation and Test in Europe*, pp. 1-6, March 2014.

[2] J. Henkel and S. Parameswaran, "Designing embedded processors - a low power perspective," *Book: Springer*, 2007.

[3] A. Alimonda, S. Carta, A. Acquaviva, A. Pisano, and L. Benini, "A feedback-based approach to dvfs in data-flow applications," *IEEE Trans. Computer-Aided Design of Integr. Circuits Systs.*, vol. 28, no. 11, pp. 1691-1704, Nov. 2009.

[4] W. Kim, M. Gupta, G.-Y. Wei, and D. Brooks, "System level analysis of fast, per-core DVFS using on-chip switching regulators," *in proc. Int'l Symp. on High-Performance Computer Architec.*, pp. 123-134, Feb. 2008.

[5] T. Kolpe, A. Zhai, and S. S. Sapatnekar, "Enabling improved power management in multicore processors through clustered DVFS," *in proc. Design Automation and Test in Europe*, pp. 1-6, March 2011.

[6] B. W. K. Wang, H. Yu and C. Zhang, "3D reconfigurable power switch network for demand-supply matching between multi-output power converters and many-core microprocessors," *in Proc. Design Automation and Test in Europe*, pp. 18-22, March 2013.

[7] M. Wens and M. Steyaert, "An 800mW fully-integrated 130nm CMOS DC-DC step-down multi-phase converter, with on-chip spiral inductors and capacitors," *in proc. Energy Conversion Congress and Exposition*, pp. 3706-3709, Sept. 2009.

[8] S. Bandyopadhyay, Y. K. Ramadass, and A. P. Chandrakasan, "20uA to 100mA DC-DC converter with 2.8 to 4.2V battery supply for portable applications in 45nm CMOS," *in proc. Int'l Solid-State Circuits Conf.*, pp. 386-387, Feb. 2011.

[9] W. Kim, D. M. Brooks, and G.-Y. Wei, "A fully-integrated 3-level DC/DC converter for nanosecond-scale DVS with fast shunt regulation," *in proc. Int'l Solid-State Circuits Conf.*, pp. 268-270, Feb. 2012.

[10] T. E. Carson, W. Heirman, and L. Eeckhout, "Sniper: Exploring the level of abstraction for scalable and accurate parallel multi-core simulation," *in proc. Supercomputing*, 2011, available at snipersim.org.

[11] "LTC3816," *available at http://www.linear.com/product/LTC3816.*

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

[13] A. A. Sinkar, H. Wang, and N. Kim, "Workload-aware voltage regulator optimization for power efficient multi-core processors," *in proc. Design Automation and Test in Europe*, pp. 1134-1137, March 2012.

[14] W. Lee, Y. Wang, D. Shin, N. Chang, and M. Pedram, "Power conversion efficiency characterization and optimization for smartphones," *in proc. of Int'l Symp. on Low Power Electronics and Design*, pp. 103-108, 2012.

[15] W. Lee, Y. Wang, D. Shin, N. Chang, and M. Pedram, "Optimizing power delivery network in a smartphone platform," *IEEE Trans. Computer-Aided Design of Integr. Circuits Systs.*, vol. 33, no. 1, pp.36-49, Jan. 2014.

[16] Y. Choi, N. Chang, and T. Kim, "DC-DC converter-aware power management for low-power embedded systems," *IEEE Trans. Computer-Aided Design of Integr. Circuits Systs.*, vol. 26, no. 8, Aug. 2007.

[17] A. Grama, G. Karypis, V. Kumar, and A. Gupta, "Introduction to parallel computing," *Book: 2nd Ed. Addison-Wesley*, 2003.

[18] S. Balakrishnan, R. Rajwar, M. Upton, and K. Lai, "The impact of performance asymmetry in emerging multicore architectures," *in proc. Int'l Symp. on Computer Architec.*, vol. 33, no. 2, pp. 506-517, 2005.

[19] C. Bienia and K. Li, "Parsec 2.0: A new benchmark suite for chip-multiprocessors," *in proc. 5th Workshop on Modeling, Benchmarking and Simulation*, June, 2009.

[20] S. C. Woo, M. Ohara, E. Torrie, J. P. Singh, and A. Gupta, "The splash-2 programs: Characterization and methodological considerations," *in proc. Int'l Symp. on Computer Architec.*, pp. 24-36, 1995.

[21] S. Musunuri and P. L. Chapman, "Optimization of CMOS transistors for low power DC-DC converters," *in proc. Power Electronics Specialists Conf.*, pp. 2151-2157, June 2005.

[22] Z. J. Shen, D. N. Okada, F. Lin, S. Anderson, and X. Cheng, "Lateral power MOSFET for megahertz-frequency, high-density DC/DC converters," *IEEE Trans. on Power Electronics*, vol. 21, no. 1, pp. 11-17, Jan. 2006.

[23] R. Erickson and D. Maksimovic, "Fundementals of power electronics," *Book: Springer, Berlin, Germany*, 2001.

[24] J. Park, D. Shin, N. Chang, and M. Pedram, "Accurate modeling and calculation of delay and energy overheads of dynamic voltage scaling in modern high-performance microprocessors," *in proc. Int'l Symp. on Low-Power Electronics and Design*, pp. 419-424, 2010.

[25] Z. J. Shen, Y. Xiong, X. Cheng, Y. Fu, and P. Kumar, "Power MOSFET switching loss analysis: A new insight," *in proc. Industry Application Conf.*, pp. 1438-1442, Oct. 2006.

[26] L. Shang, R. Dick, and N. K. Jha, "SLOPES: Hardware-software cosynthesis of low-power real-time distributed embedded systems with dynamically reconfigurable fpgas," *IEEE Trans. Computer-Aided Design of Integr. Circuits Systs.*, vol. 26, no. 3, pp. 508-526, July 2007.

[27] J. Teich, "Hardware/software codesign: The past, the present, and predicting the future," *in proc. IEEE*, vol. 100, pp. 1411-1430, May 2012.

[28] L. Balogh, "Design and application guide for high speed MOSFET gate drive circuits," *Available at http://www.ti.com/lit/ml/slup169/slup169.pdf.*

[29] T. Miller, R. Thomas, and R. T. X. Pan, "VRSync: Characterizing and eliminating synchronization-induced voltage emergencies in many-core processors," *in proc. Int'l Symp. on Computer Architec.*, pp. 249-260, June 2012.

[30] "Vishay siliconix Si4442DY datasheet," *Available at http://www.vishay.com/docs/71358/si4442dy.pdf.*

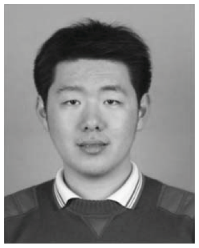
[31] "Digi-key," *Available at http://www.digikey.com/.*

[32] "Intel VRD 11.1," *available at http://www.intel.com/content/dam/doc/design-guide/voltage-regulator-down-11-1-processor-power-delivery-guidelines.pdf.*

[33] "Vishay siliconix Si4840DY datasheet," *Available at http://www.vishay.com/docs/71188/71188.pdf.*

[34] "Vishay siliconix Si4838DY datasheet," *Available at http://www.vishay.com/docs/71359/71359.pdf.*

Woojoo Lee (S'12), who received the B.S. degrees in Electrical Engineering from the Seoul National University, Seoul, Korea in 2007, and the M.S. degree in Electrical Engineering from the University of Southern California, Los Angeles, CA, in 2010. He is currently a Ph.D. candidate in Electrical Engineering at University of Southern California, under the supervision of Prof. Massoud Pedram. His research interest includes low-power VLSI design, system-level power and thermal 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, and Ph.D. degree in electrical engineering at University of Southern California, in 2014, under the supervision of Prof. Massoud Pedram. He is currently a postdoctoral research associate and (part-time) lecturer at University of Southern California. His current research interests include system-level power management, next-generation energy sources, hybrid electrical energy storage systems, near-threshold computing, digital circuits power minimization and timing analysis, cloud computing, mobile devices and smartphones, electric vehicles and hybrid electric vehicles, and the smart grid. He has published around 130 papers in these areas. He received best paper awards at 2014 IEEE International Symposium on VLSI (ISVLSI) and 2014 IEEE/ACM International Symposium on Low Power Electronics Design (ISLPED), top paper at 2014 IEEE Cloud Computing Conference (CLOUD), and best paper nominations.

Massoud Pedram (F'01), who is the Stephen and Etta Varra Professor in the Ming Hsieh department of Electrical Engineering at University of Southern California, received a Ph.D. in Electrical Engineering and Computer Sciences from the University of California, Berkeley in 1991. He holds 10 U.S. patents and has published four books, 13 book chapters, and more than 140 archival and 380 conference papers. His research ranges from 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. For this research, he and his students have received seven conference and two IEEE Transactions Best Paper Awards. Dr. Pedram is a recipient of the 1996 Presidential Early Career Award for Scientists and Engineers, a Fellow of the IEEE, an ACM Distinguished Scientist, and currently serves as the Editor-in-Chiefs of the ACM Transactions on Design Automation of Electronic Systems and the IEEE Journal on Emerging and Selected Topics in Circuits and Systems. He has also served on the technical program committee of a number of premiere conferences in his ?eld 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.