CIRCUIT AND SYSTEM LEVEL DESIGN OPTIMIZATION FOR POWER DELIVERY AND MANAGEMENT

A Dissertation

by

TONG XU

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Chair of Committee,	Peng Li
Committee Members,	Paul V. Gratz
	Rabi N. Mahapatra
	Aniruddha Datta
Head of Department,	Chanan Singh

December 2014

Major Subject: Computer Engineering

ABSTRACT

As the VLSI technology scales to the nanometer scale, power consumption has become a critical design concern of VLSI circuits. Power gating and dynamic voltage and frequency scaling (DVFS) are two effective power management techniques that are widely utilized in modern chip designs. Various design challenges merge with these power management techniques in nanometer VLSI circuits. For example, power gating introduces unique power integrity issues and trade-offs between switching noise and rush current noise. Assuring power integrity and achieving power efficiency are two highly intertwined design challenges. In addition, these trade-offs significantly vary with the supply voltage. It is difficult to use conventional power-gated power delivery networks (PDNs) to fully meet the involved conflicting design constraints while maximizing power saving and minimizing supply noise. The DVFS controller and the DC-DC power converter are two highly intertwining enablers for DVFS-based systems. However, traditional DVFS techniques treat the design optimizations of the two as separate tasks, giving rise to sub-optimal designs.

To address the above research challenges, we propose several circuit and system level design optimization techniques in this dissertation. For power-gated PDN designs, we propose systemic decoupling capacitor (decap) optimization strategies that optimally trade-off between power integrity and leakage saving. First, new global decap and re-routable decap design concepts are proposed to relax the tight interaction between power integrity and leakage power saving of power-gated PDN at a single supply voltage level. Furthermore, we propose to leverage re-routable decaps to provide flexible decap allocation structures to better suit multiple supply voltage levels. The proposed strategies are implemented in an automatic design flow for choosing optimal amount of local decaps, global decaps and re-routable decaps. The proposed techniques significantly increase leakage saving without jeopardizing power integrity. The flexible decap allocations enabled by re-routable decaps lead to optimal design trade-offs for PDNs operating with two supply voltage levels.

To improve the effectiveness of DVFS, we analyze the drawbacks of circuit-level only and policy-level only optimizations and the promising opportunities resulted from the cross-layer co-optimization of the DC-DC converter and online learning based DVFS polices. We present a cross-layer approach that optimizes transition time, area, energy overhead of the DC-DC converter along with key parameters of an online learning DVFS controller. We systematically evaluate the benefits of the proposed co-optimization strategy based on several processor architectures, namely single and dual-core processors and processors with DVFS and power gating. Our results indicate that the co-optimization can introduce noticeable additional energy saving without significant performance degradation.

DEDICATION

To my wife, Fangfang, and my parents.

ACKNOWLEDGEMENTS

This material is based upon work supported by the National Science Foundation under Grants No. 0903485 and No. 0747423 and SRC under Contract 2009-TJ-1987.

Looking back the five years of my PhD life, I have my greatest gratitude to the professors who guided me in the academic work and the friends who cheered me up.

My adviser, Dr. Peng Li, has been a great inspiration. My first thanks goes to him. I have been very fortunate to be able to work for him. He guided me to dive in the research area. With his endless support and invaluable advice, I learned the ways of how to analyze the problems and how to build concrete steps towards solving them. All of those experience added greatly to this thesis. I would also like to thank Dr. Li for always being patient and stimulating my potentials. His support is instrumental for me throughout my course of study.

Thanks also to Dr. Paul V. Gratz, Dr. Rabi N. Mahapatra, Dr. Aniruddha Datta, and Dr. Rabi N. Mahapatra who gave me suggestions and ideas of improvements of my work. The interesting discussion and helpful feedback encouraged me to pursue perfection. Their astute and challenging comments make this dissertation stronger.

Thanks for all my colleague and friends who helped me along the road, especially Boyuan Yan, Suming Lai, Zhiyu Zeng and Leyi Yin. I consider myself lucky to be able to work with them. Their dedication and enthusiasm to the work are the two most important things I have learned. I would also like to thank Dr. Li's research group. It always been enjoyable and rewarding to collaborate with them.

I would also like to thank Hai Lan, Ralf Schmitt, Savithri Sundareswaran, and Brian Mulvaney for showing me the connection between the academia and industry experience. It was a great experience of working in Rambus Inc. and Freescale

Semiconductor under the interning programs.

Finally, thanks for my wife, my parents, and my uncle's love and support. Their unconditional love and help gave me the foundation of confidence that I could insist in pursuing doctoral degree and any other challenges in my life. It is to them this work is dedicated.

TABLE OF CONTENTS

LIST OF FIGURES

FIGURE Page

LIST OF TABLES

1. INTRODUCTION

As the VLSI technology scales down, power management has become a first-order design consideration for modern chip designs including high-performance processors, embedded processors, and system-on-a-chips (SoCs). Various design issues appear with power management techniques in modern VLSI designs.

CMOS technology scaling doubles the number of transistors in a chip every eighteen months. As a cost, the power consumption of integrated circuits has dramatically grown in the past decades. The power issue of VLSI circuits becomes more critical as the technology scales down to nanometer scale. The power consumption generates a large amount of heat that may increase the temperature of circuits. Working under high temperature affects the reliability and performance of circuits. Extra cooling system and cost are required in order to dissipate the heat. In addition, battery life becomes one main concern of customers with the popularity of mobile devices. The batteries of mobile devices would be exhausted quickly by high power consumption. In order to control the power consumption, different power management techniques are proposed. During these techniques, power gating and dynamic voltage and frequency scaling (DVFS) are widely used in modern VLSI designs.

1.1 Power Management Techniques

The power consumption of an integrated circuit can be categorized into static power and dynamic power. Power gating is used to reduce the static power while DVFS is mainly used to control the dynamic power.

1.1.1 Power Gating

The static power (standby power) refers to the power consumed due to leakage current when a CMOS circuit is in standby. The percentage of chips that is idle or significantly underclocked (dark silicon) increases as the VLSI process technology scales down [3, 4]. Dark silicon is estimated to take up 20% of the chip area at the 22nm technology node and it will take up 50% at the 8nm node [5]. To this end, leakage power management becomes increasingly important for modern IC designs. Power gating is an effective solution to reduce the leakage consumption [6, 7, 8, 9, 10].

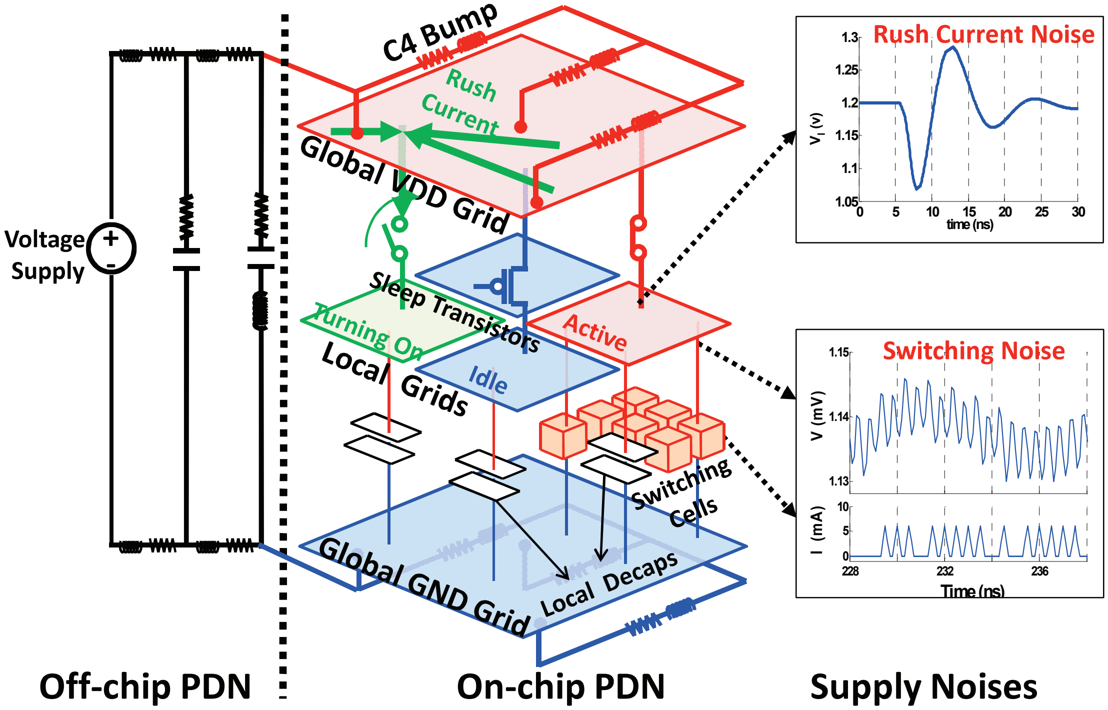


Figure 1.1: Typical structure and supply noises of power-gated PDNs. The switching noise is due to switching currents of logic devices. The rush current noise is due to rush currents created to charge up the decaps of a local grid that is woken up.

A typical power-gated power delivery network (PDN) is shown in Fig. 1.1. The

PDN is composed of an off-chip part and an on-chip part. The off-chip part includes the model of the motherboard, package, and off-chip decoupling capacitors and parasitic inductances [11]. The on-chip part includes a global VDD grid, a global GND grid and multiple local power-gated grids (power domains). Each local power grid is connected to the global V_{DD} grid through switchable sleep transistors. Hence, the leakage power can be saved by turning off the sleep transistors. Such power delivery networks have been widely adopted in multi-core chips and system-on-a-chips (SoCs) to support the power gating of multiple power domains [7, 10].

Power integrity is a significant concern in power-gated PDN designs. Two types of supply noises exist in the power-gated PDNs: switching noise and rush current (wake-up) noise as shown in Fig. 1.1. The switching noise is caused by switching activities of logic cells. When time variant switching currents flow through off-chip inductors and on-chip resistive grids, a voltage fluctuation is introduced to the logic cells. The rush current noise is a unique source of supply noise for power-gated PDNs. It is due to the rush currents that are created to charge up the decoupling capacitors in a local grid when it is woken up. The other active local grids suffer the voltage fluctuation brought by the rush currents.

The primary design challenge of a power-gated PDN stems from the conflicting objectives of power integrity and power efficiency. We summarize the key design trade-offs and typical design strategies in Fig. 1.2. The oval-shapes of the diagram indicate the design concerns and the edges indicate the typical strategies. Switching noise is typically suppressed by local decaps (LDs) that are connected between the local grids and the global GND grid [12, 13, 14]. In this case, the suppressions of switching noise and rush current noise contradict each other since local decaps are the sources of rush current noise. Hence, it is hard to achieve the power integrity by only using local decaps. Extending the turn-on time of the sleep local grid is a

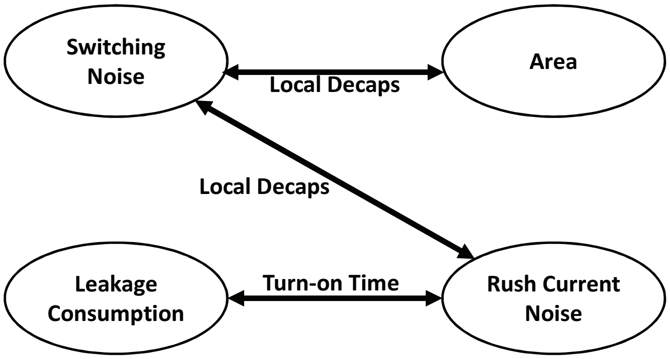


Figure 1.2: Design trade-offs and typical strategies of power-gated PDNs with a single supply voltage. The oval-shapes indicate the concerns of a PDN design. The edges indicate the typical strategies to balance design concerns.

common strategy to suppress the rush current noise [15, 16, 17]. However, longer turn-on time inevitably reduces energy saving, for there are fewer opportunities to launch power gating. As a result, the leakage saving of power gating is limited by the power integrity requirements.

Another critical problem of power-gated PDN design is the design trade-off variations under different supply voltages. Dynamic Voltage Scaling (DVS) and Dynamic Voltage and Frequency Scaling (DVFS) are widely applied to modern processors to save the dynamic power consumption. These techniques provide different supply voltages for a processor to operate at different operating points. The power-gated PDN design trade-offs between leakage saving and supply noises highly depend on the supply voltages. On one hand, the leakage current of a VLSI circuit exponentially decreases with the supply voltage (V_{DD}). Hence, power gating at lower V_{DD} requires longer break even time to compensate its energy overhead. It means that there are fewer opportunities to launch power gating at lower supply voltage. On the other hand, both switching noise and rush current noise, when normalized with respect to the nominal supply voltage, have a tendency to decrease with supply voltage. In summary, leakage saving is the dominant design concern at lower V_{DD}, while power integrity is the dominant design concern at higher V_{DD}. Fixed decap configuration is a typical strategy of power-gated PDN designs [18]. As shown in Fig. 1.3, the amount of local decaps are determined based on the switching noise at high V_{DD}, for it is the worst case of power integrity. However, this amount of local decaps is overdesigned for the power integrity at low V_{DD} since the switching noise decreases with supply voltage [19]. Obviously, the decap configuration cannot be changed once circuit design is completed. Hence, extending turn-on time becomes the only method to suppress the rush current noise at low V_{DD}. As a result, the leakage saving at low V_{DD} is restricted by the overdesigned decaps

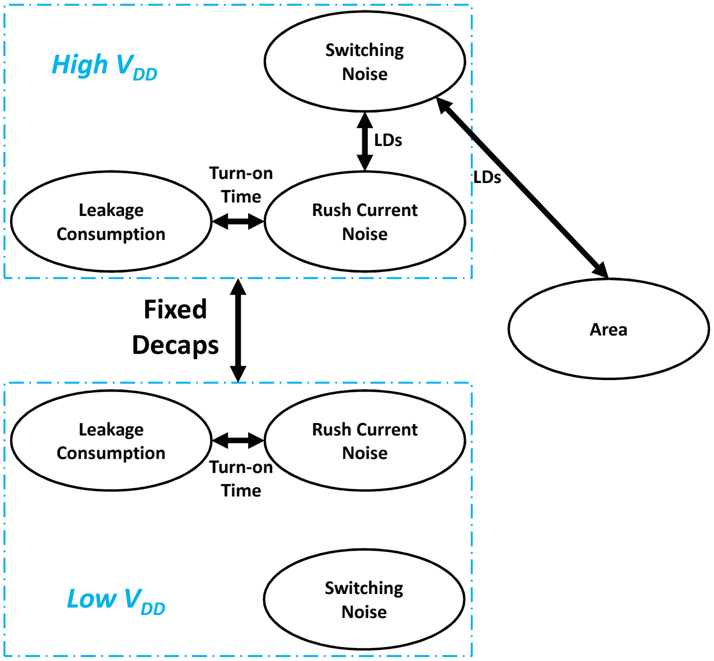


Figure 1.3: Design trade-offs and typical strategies of a power-gated PDN with two supply voltages. The oval-shapes indicate the concerns of a PDN design. The edges indicate the typical strategies to balance the design concerns.

1.1.2 Dynamic Voltage and Frequency Scaling (DVFS)

As an effective means of controlling power consumption, Dynamic Voltage and Frequency Scaling (DVFS) has been widely adopted to reduce dynamic and static power consumption [20, 7]. Supporting DVFS at the circuit level has been the subject of many circuit design works [21, 22, 19]. At the system-level, many DVFS policies have also been proposed to control power consumption by managing different operating points (voltage and frequency pairs) [23, 24]. Among these, online learning based DVFS policies have been shown to be effective for reducing chip power consumption [24].

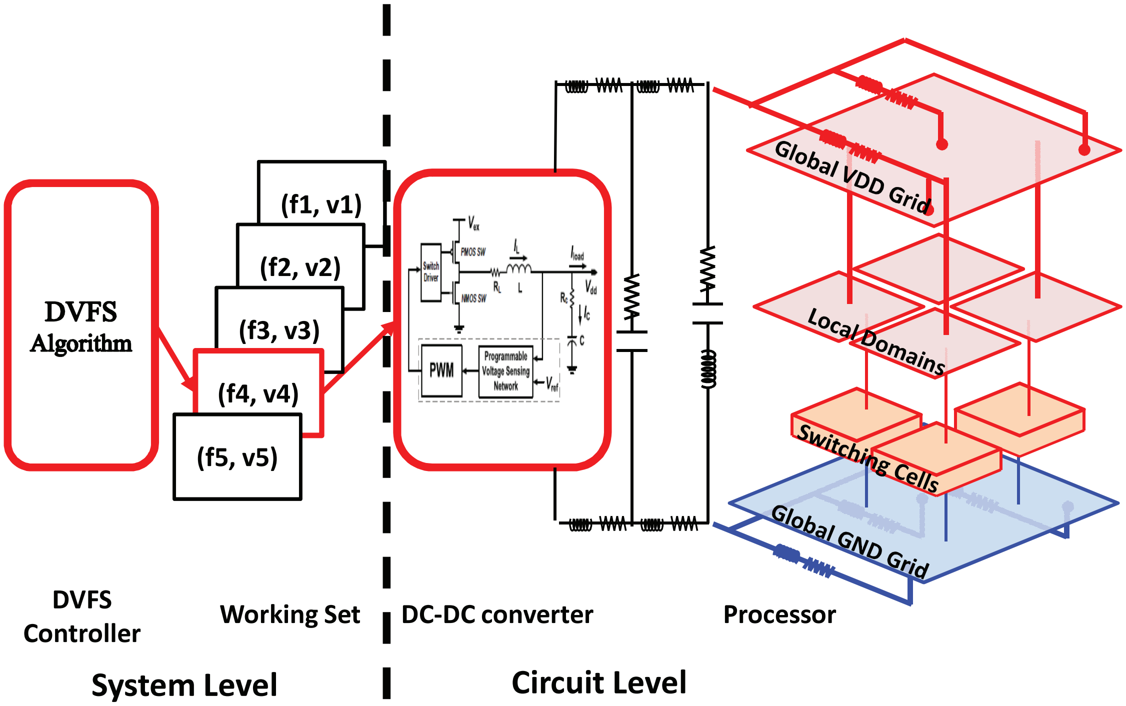


Figure 1.4: Typical structure of DVFS system that is composed of a circuit-level DC-DC converter and a system-level DVFS controller.

As shown in Fig. 1.4, a typical structure of DVFS is composed of a system-level part and a circuit-level part. At the system level, a DVFS controller is designed to balance energy consumption and performance delay by adapting to the temporal variation of workloads. The DVFS controller evaluates the energy consumption and the performance delay according to an evaluation model at the end of each operative period. Based on the evaluation, the controller selects an operating point from the working set through a control algorithm. At the circuit level, a DC-DC converter is commonly used to provide the supply voltage of the selected operating point in the following operative period.

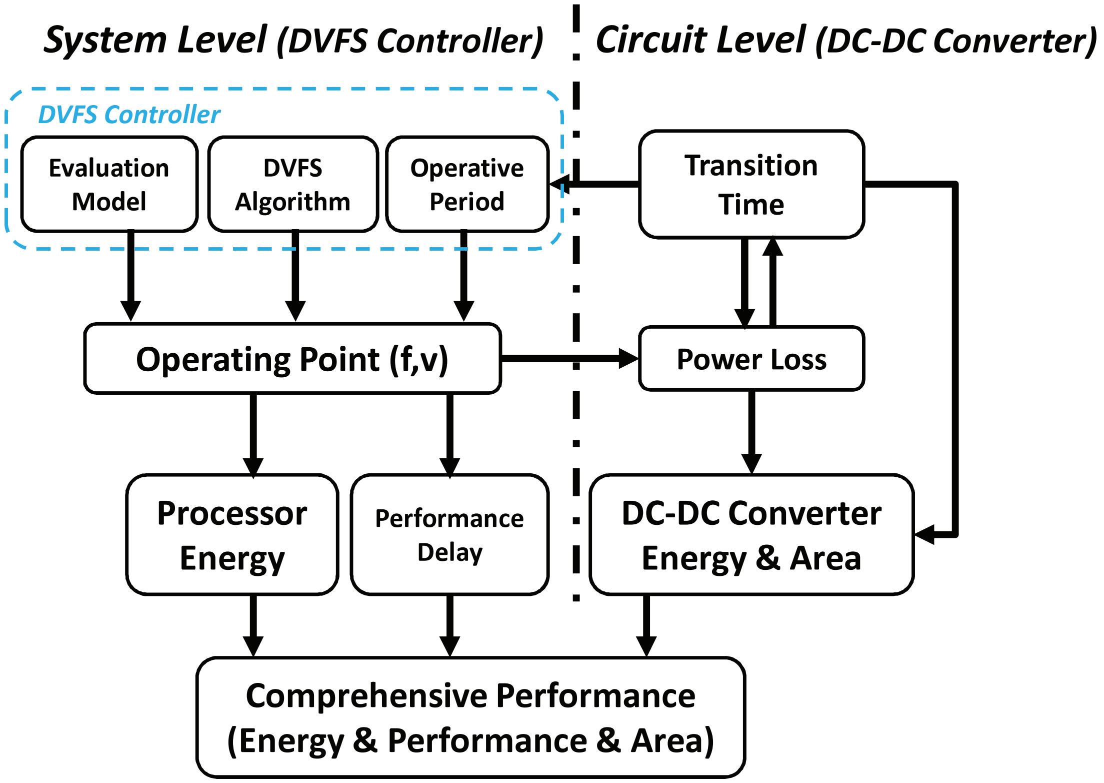

System Level (DVFS Controller)
Circuit Level (DC-DC Converter)
DVFS Controller
Evaluation Model
DVFS Algorithm
Operative Period
Transition Time
Operating Point (f,v)
Power Loss
Processor Energy
Performance Delay
DC-DC Converter Energy & Area
Comprehensive Performance (Energy & Performance & Area)

Figure 1.5: Design trade-offs of DVFS at the system level and the circuit level.

The design trade-offs of DVFS at the system level and circuit level are illustrated in Fig. 1.5. A DVFS controller can be optimized from three perspectives: operative period, evaluation model and DVFS algorithm as shown in Fig. 1.5. The operative period determines the grain of DVFS. Generally, shortening the operative period allows the DVFS controller to better track workload variations and select more suitable operating points [25, 26]. The evaluation model is used to evaluate the energy consumption and the performance delay of each operating point. Most of DVFS evaluation models are based on CPU performance counters such as cache misses, CPU usages or stall cycles [27, 28]. The accuracy of an evaluation model directly influences the performance of DVFS. A DVFS algorithm is the strategy of operating point selection. Different DVFS algorithms are proposed in the previous works [25, 29, 30, 31]. Compared with other algorithms, online learning based algorithm is more flexible to track the workload variations. In this paper, our DVFS controller adopts the online learning algorithm proposed in [24] to manage the operating points.

Nevertheless, optimizing the DVFS controller only may not necessarily lead to the minimization of the total energy consumption. First, the system-level optimization may increase the DC-DC converter's energy overhead. The fine-grained DVFS requires that the DC-DC converter supports fast transition of output voltage as shown in Fig. 1.5. However, shortening transition time may increase the DC-DC converter's energy overhead at the circuit level. In addition, the operating points selected by the DVFS policy may increase the power loss of the DC-DC converter. The DC-DC converter's energy consumption varies with its output voltage, which is well demonstrated by the power efficiency measurements in [22]. Hence, the total energy consumption may increase even if the operating points reduce the energy consumed by the CPU. Second, evaluation models that are commonly based on CPU performance counters cannot reflect the circuit-level energy consumption. As a result, a system-level optimization balances the CPU energy consumption and the performance delay, but it may not optimize the total energy consumption. Therefore, the optimizations of the DVFS controller and the DC-DC converter shall be synergistically considered as two aspects of the same power problem.

1.2 Previous Works

As mentioned in the last section, different design issues appear with power gating and DVFS. Some existing works are proposed to address these problems.

1.2.1 Previous Works of Power Gating

For power-gated PDN designs with single supply voltage, some existing works propose solutions to deal with the conflicting objectives of power integrity and power efficiency. Some works suppress the rush current noise through controlling the wake-up process [15, 32, 33, 34, 16, 35]. Stepwise turning on of sleep transistors is used in [15, 32] to suppress the rush current noise. The amount of rush current is controlled through slowing down the charging process. In [33], the authors divided logic cells into small power domains and skew the delay of sleep transistor drivers to avoid simultaneously turning on the domains. Multiple wake-up phases are proposed in [34]. The entire turning on process is partitioned into three stages. The turn-on scheme reduces the rush current during its metastable period of operation, while boosting the power supply rail when no short circuit current paths exist in the logic. However, the entire turn-on time is extended and thereby the leakage saving is reduced. Multiple sleep modes with different sleep depths are proposed in [16, 35]. Each sleep mode represents a trade-off between wake-up penalty and leakage saving through controlling the steady state potential in the sleep mode. Although the turn-on time of light sleep modes is shortened, the leakage saving of these modes is reduced correspondingly. Generally, these methods sacrifice parts of leakage saving to reduce the rush current noise. Some other works take use of extra hardware to suppress the rush current noise. The bypass power line and multi-size sleep transistors are used in [17]. But it is not economic for core-level power gating since additional global power networks is required to implement the bypass power line.

For power-gated PDN designs with multiple supply voltages, some works have been devoted on power-gated PDN design and optimization with V_{dd} higher than 1V[17, 36]. A little has been done for power gating at ultra low supply voltage. In [37], extra control circuits are proposed to suppress rush current noise in the sub-1V region. However, the different design trade-offs at higher supply voltages are not considered. In [18, 24], DVFS and power gating are combined to reduce power consumption. However, the power gating trade-offs varying between operating points are not considered.

1.2.2 Previous Works of DVFS

For most of DVFS designs, the circuit-level DC-DC converter and the system-level DVFS policy are designed separately. Various DVFS policy are proposed in [38, 39, 24, 40]. The objective of these works is to improve the DVFS controller to better track the work loads and balance the processor energy consumption and the performance delay. Different DC-DC converter designs are proposed in [41, 42, 43, 44, 45, 46, 22]. The objective of these works is to increase the energy efficiency and transition speed. However, separate designs lack the comprehensive consideration of the entire system. In this case, even if the objectives are achieved for each level, the entire DVFS system may still not reach the overall optimality.

Some existing works discuss the influence of the DC-DC converters on DVFS performance. A joint optimization of the DC-DC converter and computational core is proposed in [47] to minimize the system energy. The core architecture is improved to reduce the influence of the DC-DC converter's power loss. However, the benefit is limited in the subthreshold region (low output voltage) and the influence of the system-level management policy is not considered. The authors of [48] propose a DC-DC converter aware DVS approach, where a standard DVS algorithm is first used to determine the execution order of a set of tasks and the supply voltage for each task. Given the schedule produced by the fixed DVS algorithm, the authors optimize the DC-DC converter to minimize the system energy based on an operating-point dependent energy model of the DC-DC converter. The outcome of this circuit-level optimization leads to revision of the supply voltages while the execution order and the start time of each task are kept the same. In this approach, the system-level DVS controller is fixed and not jointly optimized with the supporting circuit. In addition, this work does not specifically target online learning based DVFS schemes which may exhibit a stronger dependency on the underlying electrical characteristics of the DC-DC converter.

As such, an interesting and practically relevant question to ask is that to what extent power management controller and DC-DC converter shall be jointly co-optimized and what benefit may be resulted from this cross-layer co-optimization. We attempt to answer this question by investigating how the performances of online-learning based the DVFS controller depend on the underlying DC-DC converter design.

1.3 Proposed Solutions

1.3.1 Proposed Solutions on Power Gating

In this dissertation, we employ both global decaps (GDs) and re-routable decaps (RDs) to deal with the design problems associated with power-gated PDNs. Fig. 1.6 shows the PDN structure proposed in this work. Global decaps are allocated between global VDD and GND grids. They are mainly used to suppress the rush current noise by providing parts of charge required by local decaps. A re-routable decap is connected to the local grid and the global VDD grid via two switches. Re-routable decaps can work as local decaps or global decaps through controlling the switches.

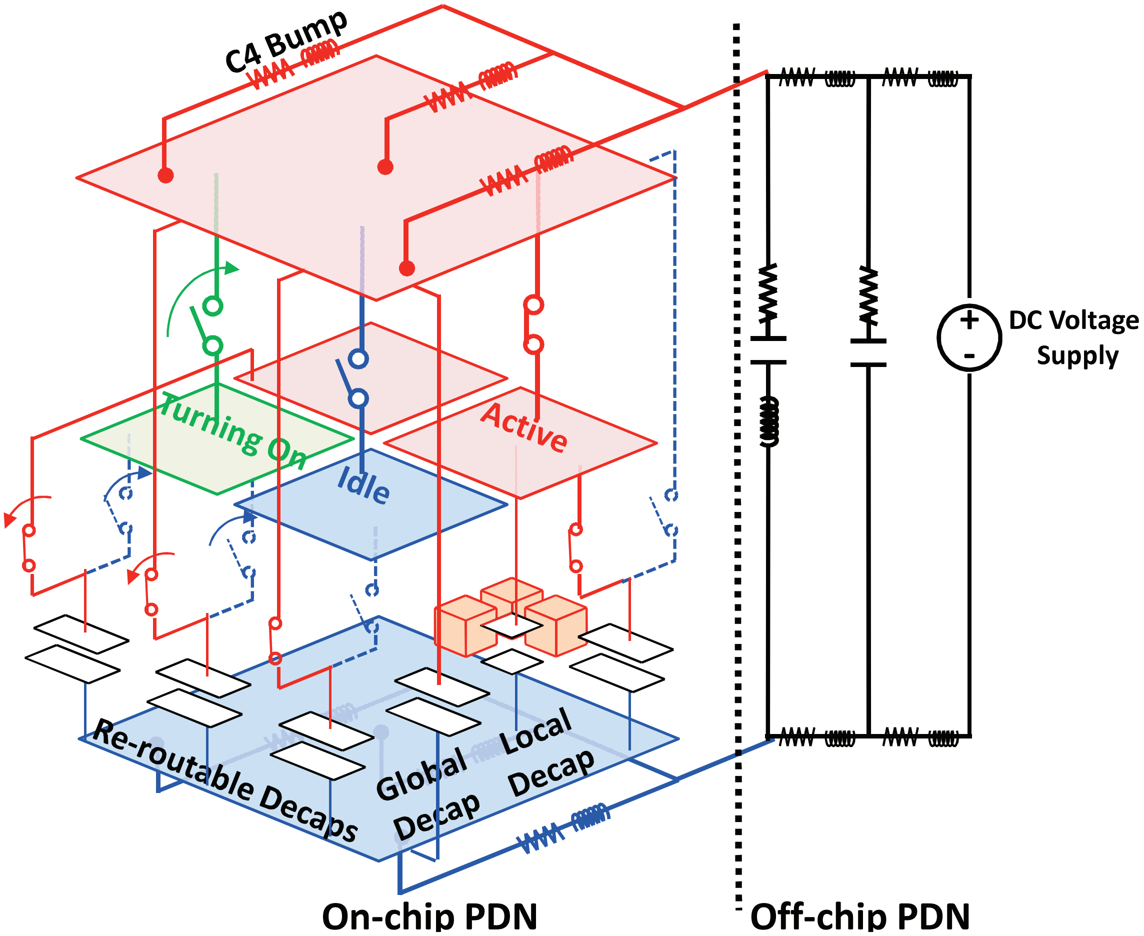


Figure 1.6: Proposed structure of power-gated PDNs. Global decaps and re-routable decaps are utilized in the proposed PDN structure.

For power-gated PDNs with single supply voltage, global decaps and re-routable decaps are utilized to relax the tight interaction between power integrity and power efficiency. As shown in Fig. 1.7, global decaps and re-routable decaps provide methods to suppress rush current noise without sacrificing the leakage saving.

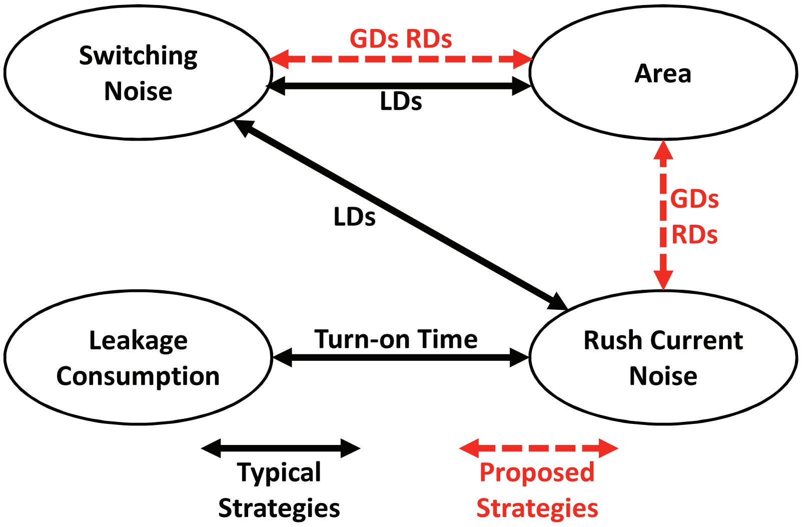


Figure 1.7: Proposed design strategies for power-gated PDN designs with a single supply voltage. The oval-shapes indicate the concerns of a PDN design. The edges indicate the strategies to balance design concerns. Black solid edges are the typical strategies. Red dash edges are the strategies proposed in this paper.

For the power-gated PDN with multiple V_{DD}, we use diverse decap configurations to adapt to supply voltage as shown in Fig. 1.8. Re-routable decaps can act as local decaps or global decaps through controlling the switches. Hence, we can provide different decap configurations (LDs/GDs/RDs) for each V_{DD} level through the utilization of re-routable decaps. In this case, the design concerns (leakage saving and power integrity) at different voltage levels can be optimized separately. Therefore, the optimal design can be achieved for each supply voltage level.

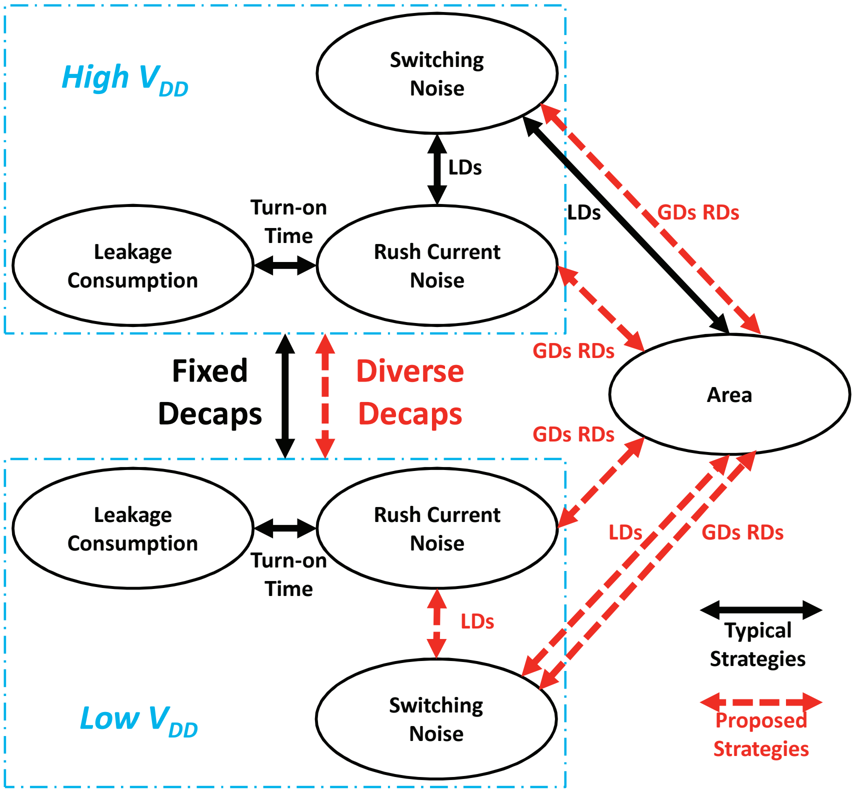


Figure 1.8: Proposed design strategies for power-gated PDN designs with multiple supply voltages. The oval-shapes indicate the concerns of a PDN design. The edges indicate the strategies to balance the design concerns. Black solid edges are the typical strategies. Red dash edges are the strategies proposed in this paper.

1.3.2 Proposed Solutions on DVFS

In this dissertation, we proceed by first analyzing design trade-offs at the circuit level and the system level respectively. Then, the interaction between the DC-DC converter design and the DVFS controller is studied. As an intermediate study, we show that performing system-level policy optimization without considering circuit-level design can lead to suboptimal power and performance trade-offs. Finally, we demonstrate the benefit of cross-layer co-optimization of online-learning based the DVFS controller and the DC-DC converter and develop a two-step design flow. In the first step, we optimize the design of DC-DC converter for power loss, output voltage transition time, and area overhead. A pareto-optimal surface of the DC-DC converter designs is created for the next step. In the second step, system-level simulation is launched to generates a series of CPU usages based on the given DVFS operative periods. The online learning DVFS controller generates a series of operating points according to the CPU usages. Based on the operating points and the power loss of the DC-DC converter, the total energy and execution time are calculated. The global optimizer updates the results and tunes circuit-level converter designs to find the optimal DVFS policy and the optimal DC-DC converter design. The proposed design strategy is evaluated based on single-core processors, dual-core processors with global DVFS, and power-gated processors with DVFS respectively. Our study shows that the co-optimization of DVFS policies and the DC-DC converter can lead to noticeable additional energy saving without significant performance degradation.

2. DECOUPLING STRATEGIES FOR POWER GATING[*]

As stated in the last chapter, special design trade-offs exit in power-gated power delivery networks (PDNs). First, the trade-off between switching noise and rush current noise determines the power integrity. The switching noise is mainly suppressed by local decaps in typical PDN designs. But local decaps are the sources of rush current noise at the same time. These two types of supply noises must be balanced carefully in order to achieve the power integrity. Second, the trade-off between rush current noise and power consumption limits the application of power gating. Rush current noise is suppressed through extending turn-on time in typical power-gated PDN designs. At the same time, long turn-on time increases the energy overhead of power gating. Hence, the power gating application is restricted to long idle time. Finally, for a power-gated PDNs with multiple supply voltages, the design concern varies with the voltage levels. In this chapter, we use global decaps and re-routable decaps to balance the trade-offs between supply noises and energy saving of power-gated PDNs. Diverse decap configurations are proposed to address the design issues of the PDNs with multiple supply voltages.

2.1 Design of Power-Gated PDNs with Single Supply Voltage

In this section, we consider the design issues associated with power-gated PDNs with single supply voltage. We first discuss the design trade-offs between power integrity and power efficiency. In order to balance the trade-offs, local/global decap

[*] Part of this chapter is reprinted with permission from "Decoupling for power gating: Sources of power noise and design strategies" by Tong Xu, Peng Li and Boyuan Yan, 2011. *In Proceedings of the 48th Design Automation Conf. (DAC)*, pages 1002−1007, Copyright[2011] by ACM.

Part of this chapter is reprinted with permission from "Design and optimization of power gating for DVFS applications" by Tong Xu and Peng Li, 2012. *In 2012 13th International Symposium on Quality Electronic Design*, pages 391−397, Copyright[2012] by IEEE.

strategy and local/global/re-routable decap strategy are proposed respectively.

2.1.1 Background

The typical structure of power-gated PDN is shown in 1.1. The PDN is composed of an off-chip package and on-chip power grids. The off-chip package includes the models of PCB and C4 bumps. The on-chip part includes a global VDD grid, a global GND grid and multiple local power-gated grids (power domains). Each local power grid is connected to the global VDD grid through switchable sleep transistors. The power gating process is implemented through controlling the sleep transistors.

Fig. 2.1 presents the process of power gating. When the local grid is busy, sleep transistors are turned on to supply the power for the local grid. The sleep transistors are turned off as soon as the idle cycles start at t_1. The supply voltage of the local grid gradually falls to 0. When the idle cycles end at t_2, the sleep transistors are turned on. It takes time $T_{on} = t_3 - t_2$ to wake up the local grid and recharge the local decaps to V_{DD}. After voltage recovery at t_3, the local grid starts to work again. The leakage consumption is saved through power gating during the power-off idle cycles $T_{idle} = t_2 - t_1$.

During the power process, the net energy saved by power gating is given as

$$E_{save} = E_{leak} - E_{over}, \tag{2.1}$$

where E_{leak} is the leakage energy saved by power gating during T_{idle} and E_{over} is the energy overhead of the power gating. The time point at which the leakage saving compensates the energy overhead ($E_{leak} = E_{over}$) is the break even point t_{BEP}. The break even time is defined as $T_{BE} = t_{BEP} - t_1$. If $T_{idle} < T_{BE}$, the energy overhead overwhelms the leakage saving and thereby the power gating should not be applied to the idle time slot.

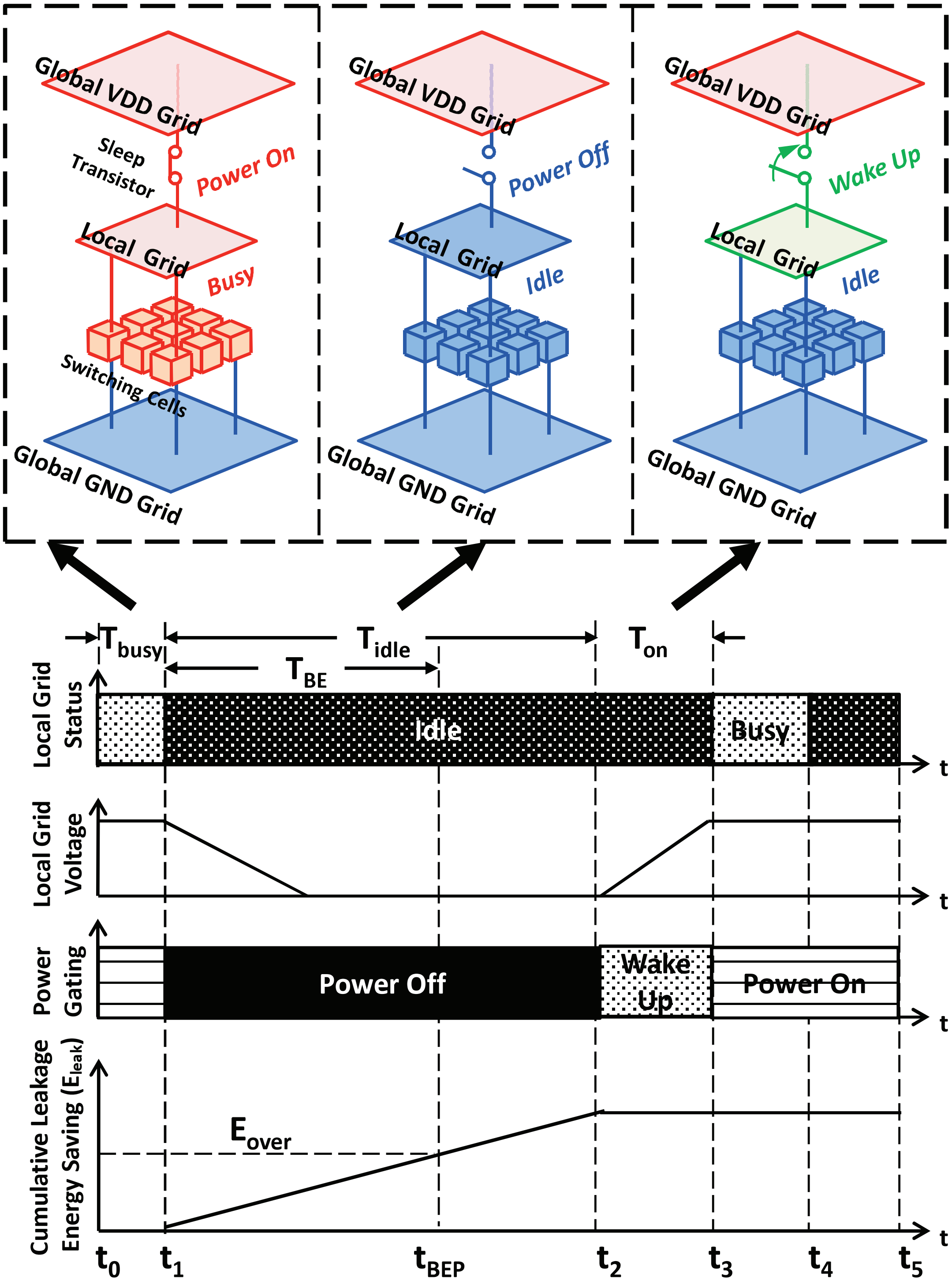


Figure 2.1: Schematic of power gating process. The sleep transistor is supposed to be turned off as soon as the idle cycles arrive. t_{BEP} is the break even point at which the energy saving compensates the energy overhead ($E_{leak} = E_{over}$).

Supply noises suppression is an important design concerns of power-gated PDNs. Switching noise and rush current noise are the two types of supply noises appear during the power gating process.

In typical power-gated PDN designs, local decaps are mainly used to suppress the switching noise [12, 13, 14, 49, 50]. There works improve the efficiency of on-chip decaps from the perspective of distribution, material or structure. However, the decaps are mainly allocated on local grids to suppress switching noise. They are not effectively used to reduce rush current noise. In contrast, local decaps are the sources of rush current noise during the process of wake up.

The rush current noise is typically suppressed through extending turn-on time T_{on} [15, 32, 33, 34, 16, 17]. Stepwise wake-up techniques are proposed in [15, 32]. These techniques turn on the sleep transistor in stepwise manners. The stepwise wake-up process can be implemented in either by dynamically controlling the gate-to-source voltage of a sleep transistor or by turning on only a portion of the sleep transistor at one time. A local grid is slowly turned on until the drain-to-source voltage of sleep transistors are significantly reduced. Then the local grid is turned on completely until the voltage recovers to V_{DD}. The peak of rush current is controlled in safe range by these techniques. In [33], the rush current is reduced through turning on the local domains at different time. The logic cells are divided into small power domains. The sleep transistors of these domains are driven by a driver tree. The local domains are turned on at different time through slewing the delay of sleep transistor drivers. Multiple wake-up phases are proposed in [34]. The entire turning on process is partitioned into three stages. The turn-on scheme reduces the rush current during its metastable period of operation, while boosting the power supply rail when no short circuit current paths exist in the logic. However, the entire turn-on time is extended and thereby the leakage saving is reduced. Multiple sleep modes with different sleep depths are proposed in [16, 35]. In the detest sleep mode, the local grid is completely turned off during idle time. For the other sleep modes, the voltage of local grid falls to a certain level after power gating. In this case, a local grid in light sleep mode can be waken up quickly with small rush current. Although the rush current of light sleep modes is reduced, the leakage saving of these modes is reduced correspondingly. Generally, these works suppress the rush current noise through extending the turn-on time. However, long turn-on time may reduce the opportunities of power gating and thereby reduce the leakage saving. In addition, the performance delay is increased due to the long turn-on time.

In order to address these drawbacks, we analyze the design concerns and trade-offs systematically in the following section.

2.1.2 Design Concerns and Trade-Offs

As discussed in the last section, power integrity and power efficiency are the two important design concerns of power-gated PDNs. In this section, we discuss the trade-offs among switching noise, rush current noise and energy saving.

2.1.2.1 Trade-Off between Switching Noise and Rush Current Noise

Switching noise and rush current noise are the two types of supply noises associated with power-gated PDNs. The power integrity of logic devices depends on the superposition of these two types of supply noises.

Switching noise appears when a local grid is powered on. As shown in Fig. 2.2, a time variant switching current is created when a logic device is active. The voltage of device fluctuates as the switching current flows through the power grids. Logic errors may happen under the influence of voltage drop that is called as switching noise. As shown in Fig. 2.3, the switching noise is composed of high-frequency and mid-frequency components. The high-frequency component is due to the IR drop

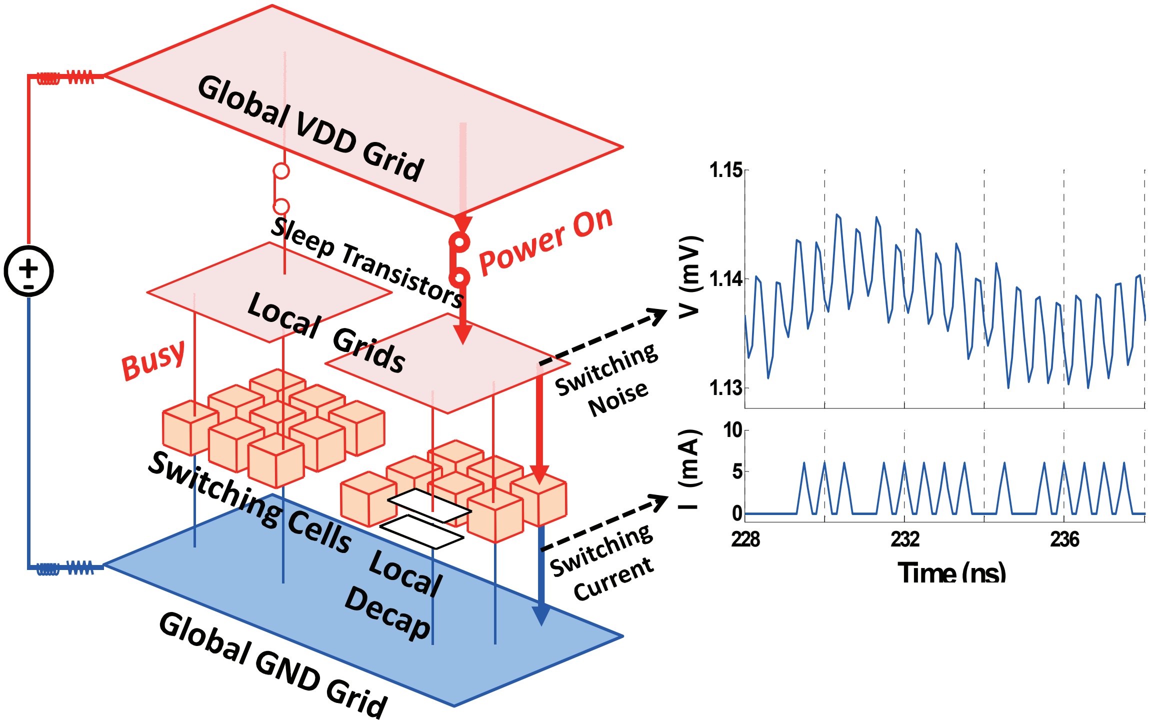


Figure 2.2: Switching noise of power-gated PDN. The switching noise appears when the local grids are powered on. The switching noise is created by the switching current of logic devices.

caused by resistive power grids. While the mid-frequency component is due to the resonance from the on-chip capacitance and the package inductance. Switching noise is typically suppressed by local decaps that are connected between the local grids and the global GND grid. First, local decaps can provide parts of the current required by nearby switching devices. Hence, they are effective to suppress the high-frequency component of switching noise. In addition, local decaps reduce the peak impendence of the PDN through providing low impendence paths. Hence, they are effective to suppress the mid-frequency component of switching noise as well.

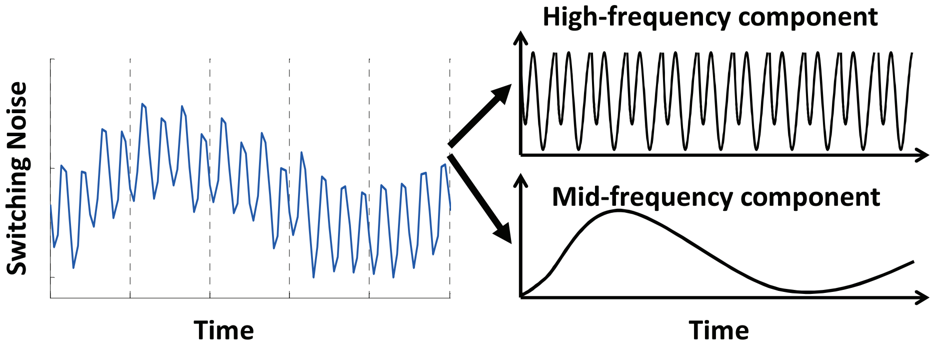


Figure 2.3: Components of switching noise. The high-frequency component is due to the IR drop on resistive power grids. The mid-frequency component is due to the resonance from the on-chip capacitance and the package inductance.

The rush current noise appears during the wake up process. The voltages of local decaps fall to 0 after the local grid is turned off. In the wake up process, a rush current is created to charge up the decaps in the local grid as shown in Fig. 2.4. The rush current flows through the power grids and decrease the voltage of the power grid. Hence, the other active local grids may suffer the rush current noise and generate logic errors. It can be seen that local decaps are the primary sources of rush current noise. Extending the turn-on time is a common method to reduce rush current noise in typical PDN designs. Turn-on time is related with leakage saving and performance delay that is discussed in the following section.

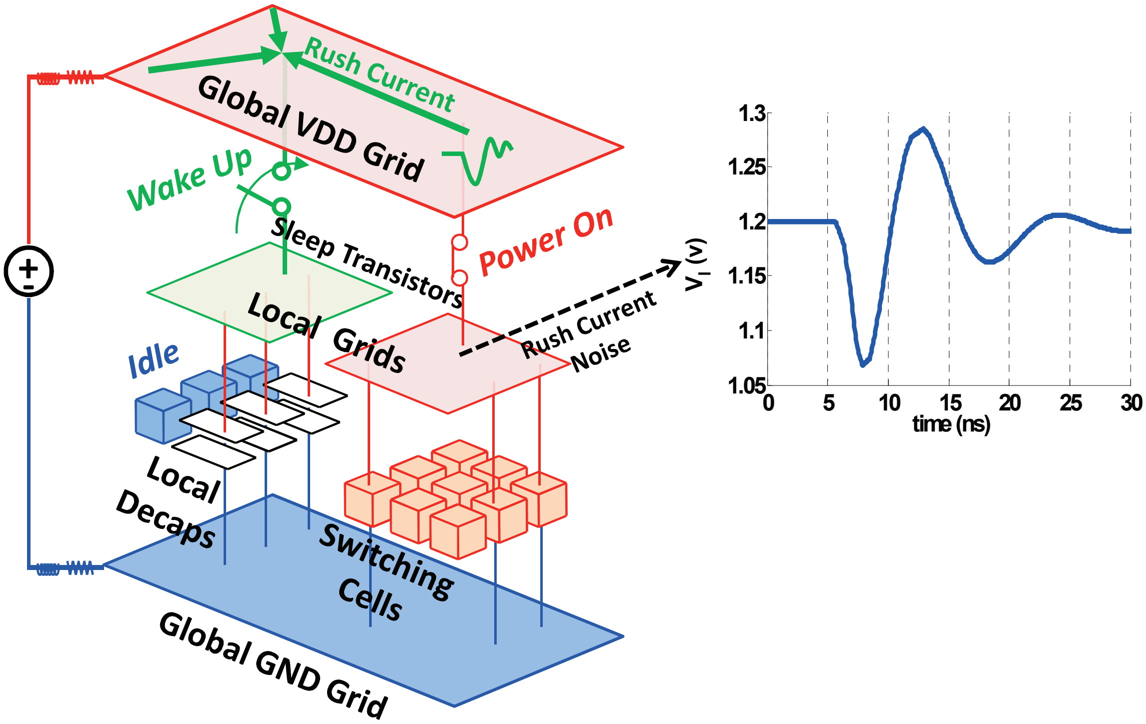


Figure 2.4: Rush current noise of power-gated PDN. Rush current noise appears during the wake up process. Rush current noise is due to the rush current created to charge the local decaps.

2.1.2.2 Trade-Off between Rush Current Noise, Energy and Performance

We introduce the process of power gating in Section 2.1.1. Power gating saves the leakage consumption during the idle cycles $T_{idle} = t_2 - t_1$ shown in Fig. 2.1. But the benefit obtained is at the cost of the performance delay and the energy overhead.

The total execution time for a single task without power gating is $T_{idle} + T_{busy}$. With power gating, the total execution time is extended to $T_{idle} + T_{busy} + T_{on}$. Therefore, the turn-on time T_{on} is the performance delay of the power gating technique.

The energy overhead of power gating is given as

$$E_{over} = E_{ctrl} + E_{LD} + E_{on}, \tag{2.2}$$

where E_{ctrl} indicates the energy spent on sleep transistor controlling, E_{LD} is the energy consumed to recharge the local decaps and E_{on} is the leakage energy consumption during turn-on time T_{on}. The time during which the leakage saving compensates the energy overhead ($E_{leak} = E_{over}$) is the break even time T_{BE}. If $T_{idle} < T_{BE}$, the energy overhead overwhelms the leakage saving and thereby the power gating should not be applied to the idle time slot. For example, the idle slot from t_4 to t_5 in Fig. 2.1 is too short to save energy through power gating. Hence, lots of leakage saving opportunities are missed due to the energy overhead (E_{over}).

Turn-on time plays a key role in determining the trade-offs between energy saving, performance delay, and rush current noise. Shortening turn-on time reduces energy overhead (E_{on}) and performance overhead (T_{on}). But, in order to reduce rush current noise, turn-on time is increased so that LDs are charged slowly thereby reducing rush current noise. An increase in turn-on time can eat into the leakage savings obtained through power gating.

2.1.3 Proposed Local/Global Decap Strategy

As discussed previously, rush current noise is mainly suppressed by extending the turn-on time. However, long turn-on time may reduce the power gating opportunities and increase the performance delay. In this section we propose local/global decap strategy (LD&GD strategy) to further reduce supply noises especially the rush current noise. A global decap is connected between the global VDD grid and the global GND grid as shown in Fig. 2.5. With the utilization of global decaps, more energy can be saved through shortening the turn-on time.

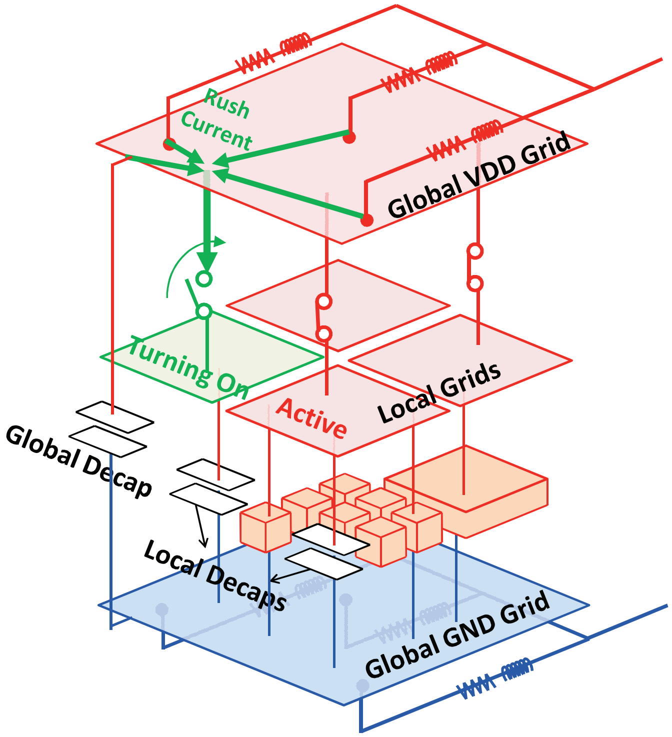


Figure 2.5: Structure of global decaps. Global decaps are allocated between the global VDD grid and the global GND grid. The main utilization of global decaps is to suppress rush current noise through providing parts of charging current during the wake up of local grid.

2.1.3.1 Switching Noise Suppression

The LD&GD strategy utilizes both local decaps and global decaps to suppress switching noise. Global decaps are able to suppress switching noise (both high- and mid-frequency components), though they are not as efficient as equal amount of local decaps.

The schematic layout and the top view of a typical PDN with a local decap is shown in Fig. 2.6. The schematic layout is based on a real industrial processor

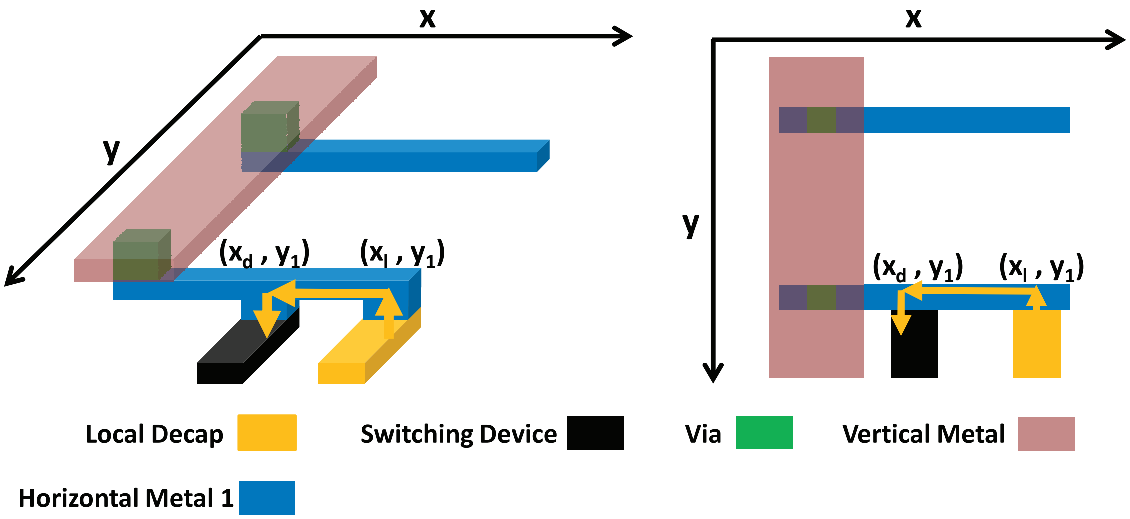


Figure 2.6: The schematic layout and the top view of a typical PDN with a local decap. Only the global VDD grid and the local grid are shown in the figure. The global GND grid is not depicted in the figure.

design with standard cells. The local grids are implemented by horizontal metal layer 1 (M_{H1}) and vertical metal layer (M_V). The local decap is located in the same row of the switching cell. The resistance between the local decap and the switching cell is a short metal segment on M_{H1}. Hence, the local decap can effectively suppress the high-frequency component of switching noise due to the small RC delay.

The schematic layout of a global decap is shown in Fig. 2.7. The global grids are composed of horizontal metal layer 2 (M_{H2}) and vertical metal layer (M_V). M_{H1} and M_{H2} are connected by the cell of a sleep transistor. The resistance between the global decap and the switching cell is composed of the resistance of global grid (metal wires and vias), the equivalent resistance of the sleep transistor, and the resistance of local grid (metal wires and vias). The high resistance path introduces a large RC delay. Hence, the global decap is not as efficient as a local decap to suppress the high-frequency switching noise.

Global decaps are also able to suppress the mid-frequency component of switching

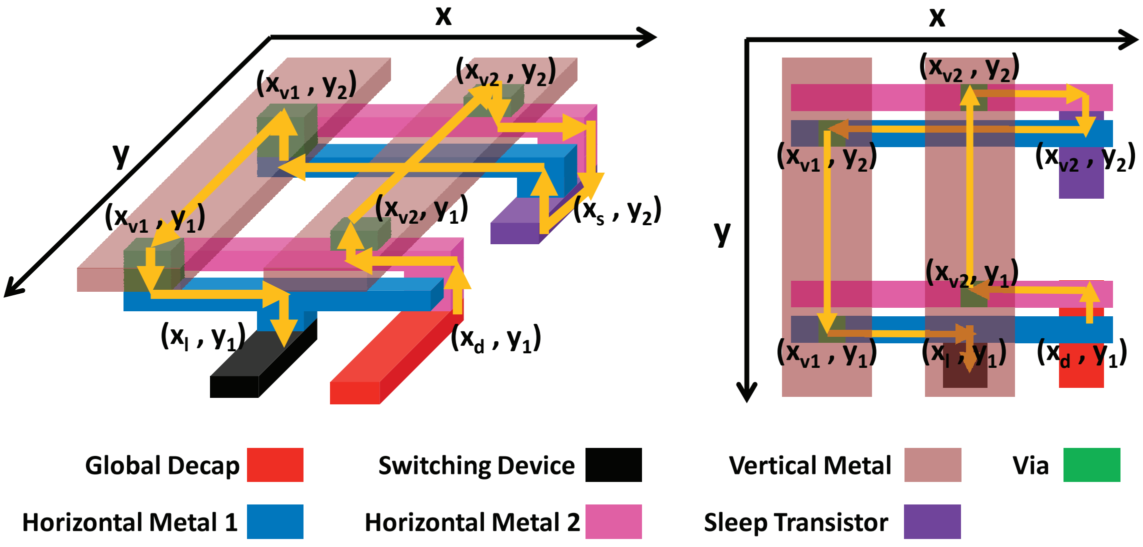
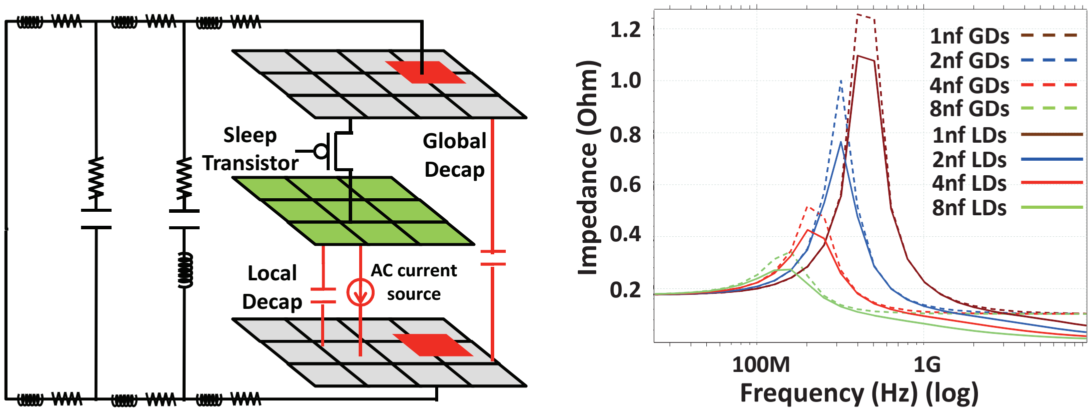


Figure 2.7: The schematic layout and the top view of a PDN with a sleep transistor and a global decap. Only the global VDD grid and the local grid are shown in the figure. The global GND grid is not depicted in the layout. Horizontal metal layer 1 and 2 are connected by a sleep transistor.



(a)

(b)

Figure 2.8: On-chip decaps' influence on circuit resonance at 45nm technology node. (a) The circuit model for analysis. (b) Impedances of the chip with different amount of local decaps and global decaps.

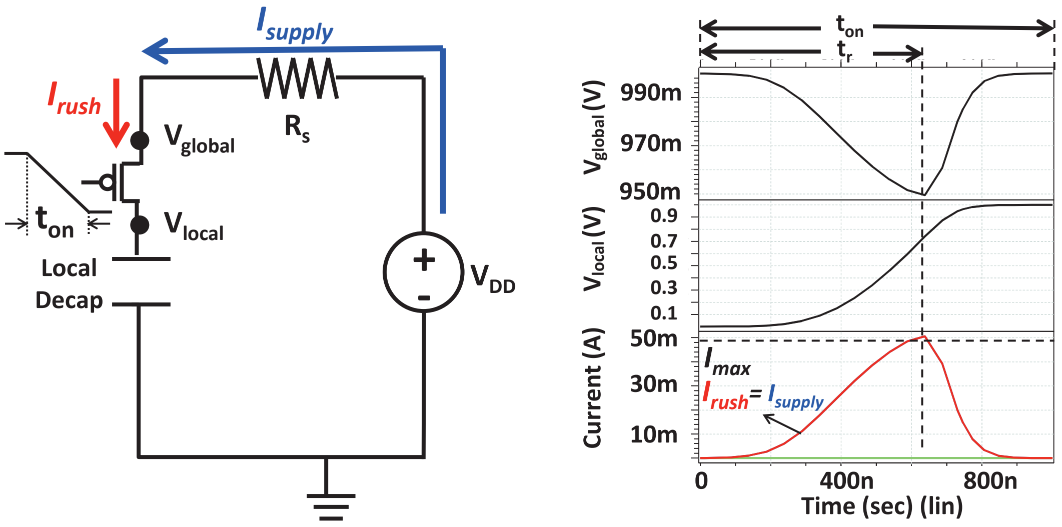
noise. The mid-frequency switching noise is due to the resonance of the circuit that can be measured in the circuit shown in Fig. 2.8(a). In this circuit, the DC supply voltage of the circuit is shorted. All the current loadings are removed. Only one AC current source is connected with the power grid. The amplitude of the current source is 1A. In this case, the impedance looking from the current source is shown in Fig. 2.8(b). We compare the impedance with local decaps and the impedance with the same amount of global decaps. Both local decaps and global decaps are able to suppress the peak of the resonance. However, the resonance reduction with local decaps is more obvious than the one with the same amount of global decaps. This is because the resistance between the local decaps and the current loading is much smaller and thereby they can provide a lower impedance path.

Although global decaps have the ability to suppress the switching noise, they are not as efficient as equal amount of local decaps. Therefore, local decaps are still the main technique to suppress the switching noise in our proposed PDN design.

2.1.3.2 Rush Current Noise Suppression

When a local grid is turned on, a rush current is created to charge the local decaps on that local grid. The local decaps include no-switching logic cells that act as capacitors and decoupling capacitance cells. The rush current leads to voltage drops in the global grid and the other active local grids. As a result, logic devices on the other active local grids may generate logic errors due to the voltage drops (rush current noise). The LD&GD Strategy takes use of global decaps to reduce the rush current and thereby turn-on time can be further shortened to save more energy.

Extending the turn-on time T_{on} suppresses the noise by decreasing the peak of rush current. Sleep transistors and the local decaps are modeled as the source of the rush current as shown in the simple circuit example of Fig. 2.9(a). R_s is the



(a) (b)

Figure 2.9: Rush current noise suppression through extending turn-on time t_{on} at 45nm technology node. (a) Simple circuit model with no global decap. R_s indicates the equivalent resistance between the supply voltage and the sleep transistor. (b) Voltage drop observed of global grid and the corresponding rush current.

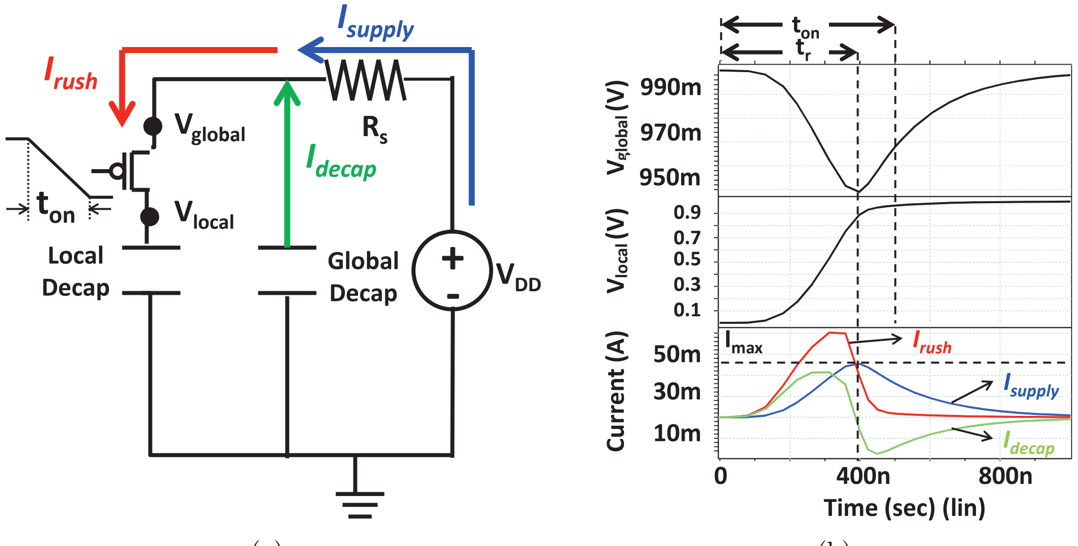
equivalent resistance between the supply voltage and the sleep transistor. $I_{supply}(t)$ is the current provided by the supply voltage. $I_{rush}(t)$ is the rush current drawn by the sleep transistor. In order to achieve the power integrity, the current provided by the power supply must meet

$$\begin{aligned} I_{supply}(t) &\leq I_{max} \\ &= \frac{r \times V_{DD}}{R_s}, \end{aligned} \tag{2.3}$$

where I_{max} is the upper bound of I_{supply} without power integrity violation, r is the ratio of the maximum tolerable rush current noise to the supply voltage. In a typical PDN design, power supply is the only source to provide the rush current. Hence, we have

$$I_{rush}(t) = I_{supply}(t). \tag{2.4}$$

In this case, the turn-on time of the sleep transistor (T_{on}) must be long enough to make sure the peak of the rush current $I_{rush}^{peak} \leq I_{max}$ as shown in Fig. 2.9(b). The voltage of the local grid/decap V_{local} takes longer time to recover to V_{DD} since the charging process is slowed down.



(a)

(b)

Figure 2.10: Rush current noise suppression of global decaps. The technology node is 45nm. (a) Simple circuit model with a global decap. R_s indicates the equivalent resistance between the supply voltage and the sleep transistor. (b) Voltage drop observed of global grid and the corresponding rush current.

To this end, without extending the turn-on time, global decaps can be used to suppress the noise by reducing I_{supply} instead of I_{rush}. As shown in Fig. 2.10(a), both the power supply and the global decap are the sources to provide charging current.

Hence, the rush current is given as

$$I_{rush}(t) = I_{supply}(t) + I_{decap}(t), \tag{2.5}$$

where $I_{decap}(t)$ is the current provided by the global decap. With the charge from the global decap, it is not necessary to slow down the charging process in order to guarantee 2.3. Therefore, the voltage of the local grid can rise to V_{DD} quickly.

The utilization of global decaps relaxes the constraint of turn-on time. The turn-on time can be significantly shortened since the rush current noise is reduced by the global decaps.

2.1.3.3 Design Strategy

According to the analysis above, the impacts of local decaps (LDs), global decaps (GDs), and turn-on time (T_{on}) on the design concerns are summarized in Tab. 2.1. Based on these impacts, the LD&GD design strategy uses local decaps and global decaps to suppress switching noise and rush current noise respectively. After the power integrity specification is met, turn-on time is further shortened to apply power gating for shorter idle time, reduce the energy overhead and thereby save more leakage power.

Table 2.1: Impacts of On-Chip Decaps and Turn-on Time on Design Concerns

Design Option	Switching Noise	Rush Current Noise	Power Gating Opportunities	Energy Overhead	Execution Time
LD Insertion	↓	↑	↓	↑	−
GD Insertion	↘	↓	−	−	−
T_{on} Shortening	−	↑	↑	↓	↓

↑increase ↓decrease ↘slightly decrease −no change

The power integrity specification may be specified as follows. First, total supply noise (superposition of switching noise and rush current noise) should be smaller than the maximum tolerable voltage drop. Second, switching noise and rush current noise should be respectively smaller then their own tolerance. In practice, one may set up a tighter tolerance for one of the two noises, say, rush current noise, as it may lead to an overall smaller budget for decoupling capacitance. In practice, the total decaps budget is limited due to fixed on-chip white space. Therefore, the total supply noise and each type of noises is tuned by the proportion between local decaps and global decaps. In Fig. 2.11, the total decap budget (100nf) is divided into local decaps and global decaps. Rush current noise is reduced though increasing the ratio of GDs to LDs, while switching noise is reduced by decreasing the ratio.

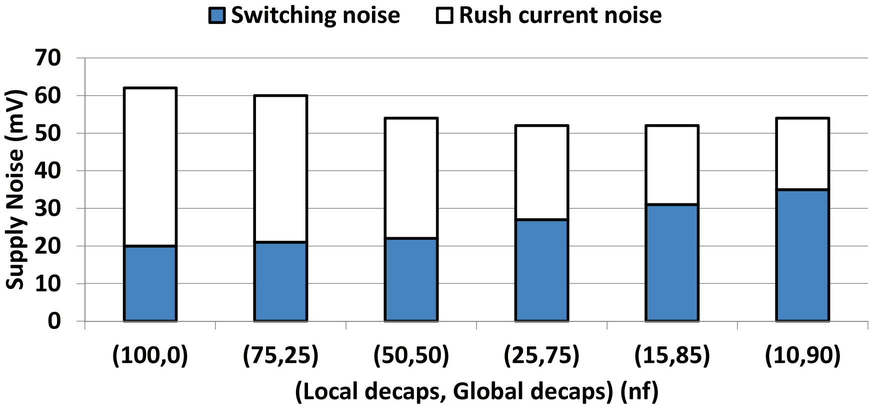


Figure 2.11: Trade-off between switching noise and rush current noise. The power-gated PDN utilized for simulation is shown in Fig. 1.6. Total decap budget (100nf) is divided into local decaps and global decaps. Local decaps and global decaps are uniformly distributed on local grids or global grids. The switching devices are modeled as triangular current sources [1]. Turn-on time is 1000ns. The technology node is 45nm.

Besides the decap configuration, turn-on time is another design parameter that determines the total supply noise. As shown in Fig. 2.12, with the use of global decaps, the proposed LD&GD Strategy is to exploit an optimal split between LDs and GDs for a given total decap budget to adjust the ratio between rush current noise and switching noise and maximize the overall power integrity.

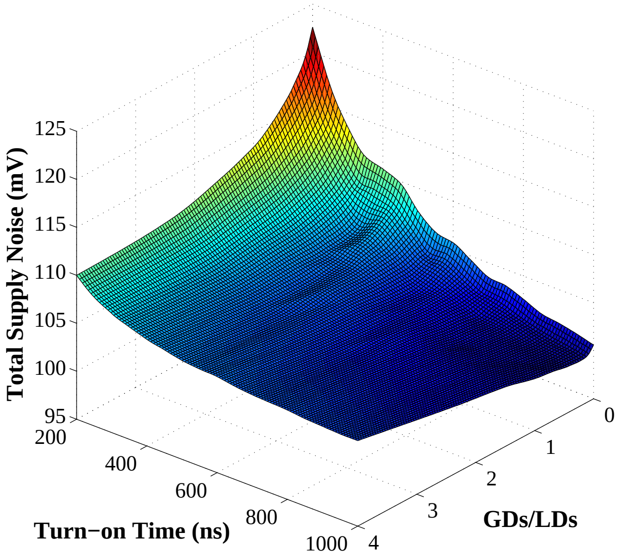


Figure 2.12: Total supply noise is controlled though the LD&GD design strategy. Total decap budget (100nf) is divided into local decaps and global decaps. Local decaps and global decaps are uniformly distributed on local grids or global grids. The switching devices are modeled as triangular current sources [1]. The technology node is 45nm.

The drawback of LD&GD design strategy is that a large amount of global decaps is needed. Assume that the maximum tolerable voltage droop is $0.1V_{DD}$, the total charge to re-charge the local decaps is given by

$$Q_{rush} = 0.9V_{DD}C_{local}, \tag{2.6}$$

where C_{local} is the amount of local decaps. If all the charge is provided by the global decaps, we need approximately

$$C_{global} = \frac{Q_{rush}}{0.1V_{DD}} = 9C_{local}. \tag{2.7}$$

In most of the cases, it is hard to meet this requirement of global decaps.

2.1.4 Proposed Local/Global/Re-Routable Decap Strategy

As discussed in the previous section, large amounts of decaps are needed in order to achieve a short turn-on time. It can be very hard to find a feasible decap configuration when the decap budget is very limited. To deal with this problem, we propose the Local/Global/Re-routable Decap Strategy (LD&GD&RD strategy) that uses re-routable decaps (RDs), a new design concept proposed in our recent work [36], to further relax the tight interaction between power integrity and power leakage saving.

2.1.4.1 Structure and Functions of Re-Routable Decaps

The structure of re-routable decaps is shown in Fig. 2.13. Re-routable decaps are essentially programmable decoupling devices. For each re-routable decap, two switches S_L and S_R are used to control the decap routing. The functionalities of re-routable decaps are described below.

Function 1

The first function of a re-routable decap is to act as a local decap for its own local grid as shown in Fig. 2.13. When the local grid is active, S_R is off and S_L is on. The re-routable decap is connected to local grids as a local decap to suppress switching noise. The equivalent resistance of S_L can impact the efficiency of a re-routable decap to suppress switching noise. Next section discusses design requirements for S_L

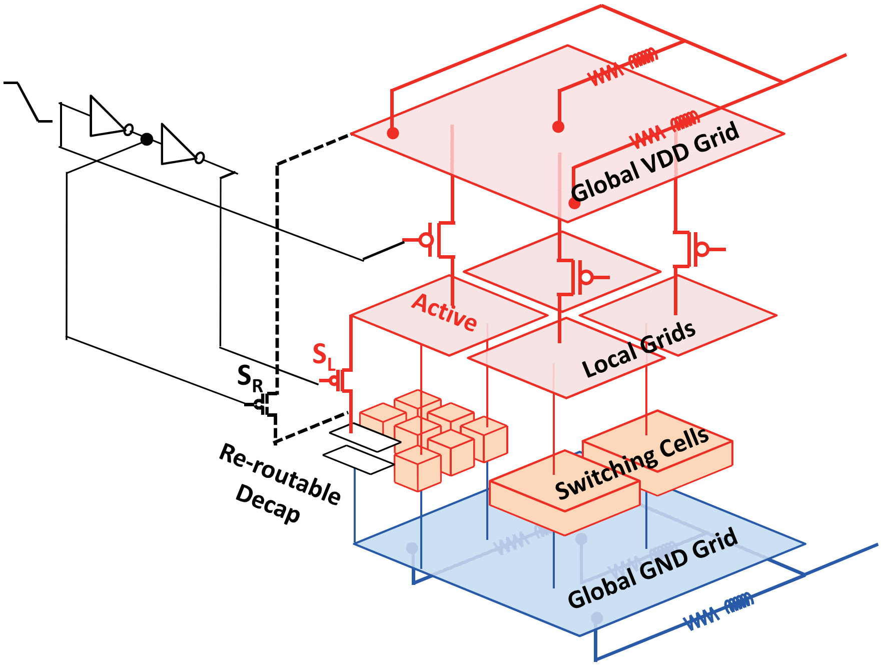


Figure 2.13: Re-routable decap Function 1: when the local grid is active, the re-routable decap acts as a local decap to suppress the switching noise of its own power domain.

Function 2

The second function of re-routable decap is to act as a global decap and preserve the charge on itself as shown in Fig. 2.14. When local grid A goes to sleep, S_L is turned off and S_R is turned on. The re-routable decap is routed to the global VDD grid. During this time, it acts as a global decap that aids in suppressing both switching noise on global grid and rush current noise on neighboring local grids For example, local grid B creates rush current during its wake up process. The rush current brings rush current noise to active local grid C and D. The re-routable decap provides current required by local grid B and thereby reduces the rush current noise of C and D. Most of the charge on re-routable decaps is preserved by the global VDD grid. Hence, when the re-routable decap is routed back to local grid A (Function

1), it creates much less rush current noise than a local decap during A's wake up process. As a result, the rush current noise created by local grid A is significantly reduced.

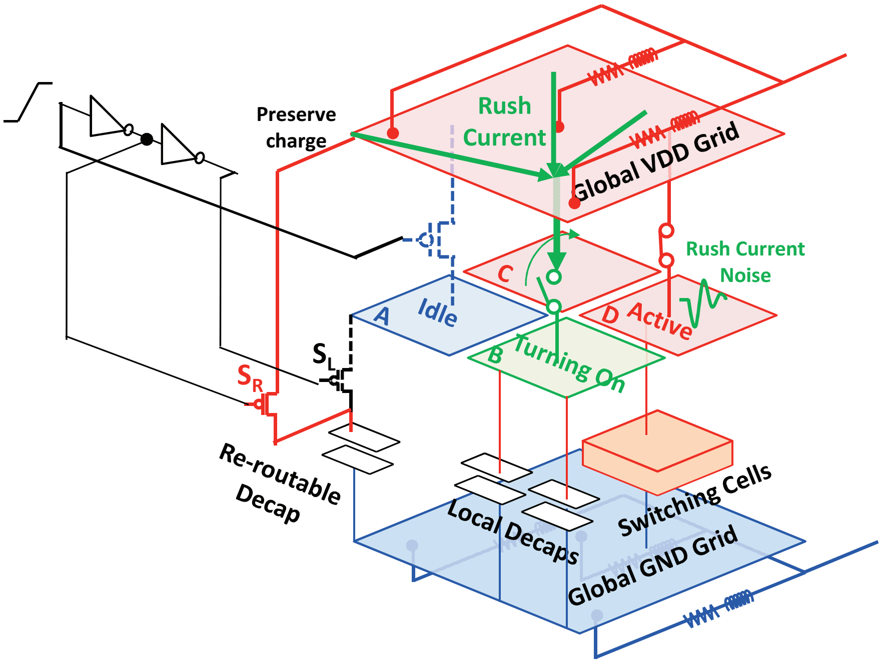


Figure 2.14: Re-routable decap Function 2: when the local grid is turned off, the re-routable decap is routed to the global VDD grid. It acts as a global decap to suppress the supply noises of other local domains. In addition, the significant charge on re-routable decap is preserved by the global grid.

A special case is that the power integrity specification of a local grid can be met by GDs and its own LDs and RDs. In this case, it is not necessary to have re-routable decaps work as global decaps to suppress the supply noises of other active local grids. Hence, S_R is only used to preserve charge on the re-routable decap. Since only leakage current flows through S_R, it can be made small to reduce the area overhead.

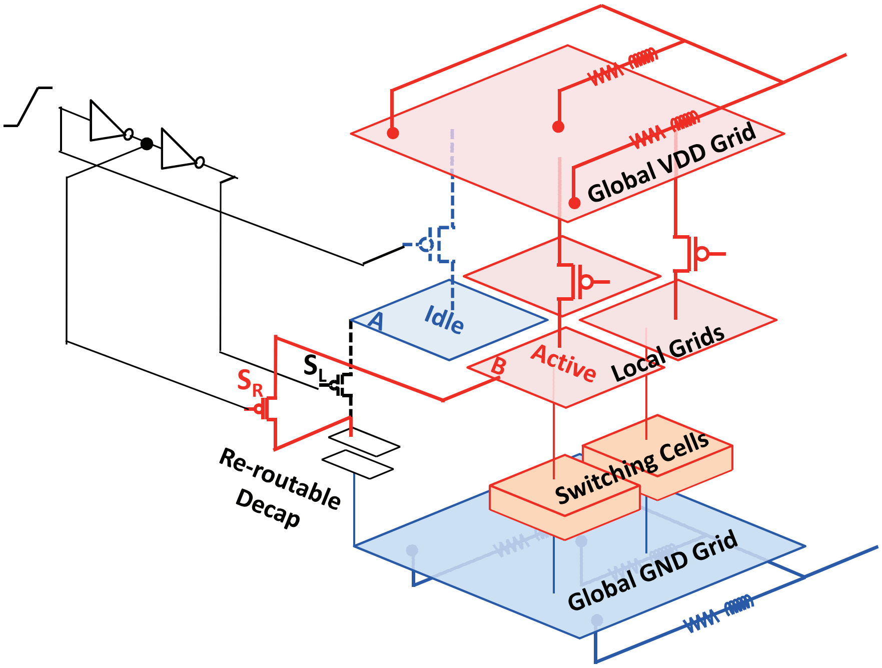


Figure 2.15: Re-routable decap Function 3: when the local grid is turned off, the re-routable decap is routed to the other active local grids. It acts as a local decap to suppress the switching noises of other local domains.

Function 3

Another function of re-routable decap is to act as a local decap of an active local grid as shown in Fig. 2.15. When local grid A goes to sleep, S_L is turned off and S_R is turned on. The re-routable decap is routed to active local grid B. It acts as a local decap to suppress the switching noise of local grid B. This function can be used for workload variations. For example, the workloads are assigned to local domain A and B. The workloads of B may increase after A is turned off. In this case, local domain B need more local decaps to suppress the additional switching noise.

In this dissertation, we focus on the applications of Function 1 and 2.

2.1.4.2 Advantages of Re-Routable Decaps

We summarize and compare different types of on-chip decaps in Tab. 2.2. Re-routable decaps avoid the disadvantages of LDs and GDs. First, re-routable decaps are more efficient than global decaps to suppress the switching noise. Compared with global decaps, re-routable decaps are allocated on the same metal layer of the switching cells. Hence, they are closer to the sources of switching noise than global decaps. Second, re-routable decaps reduce rush current and energy overhead of power gating. The charge of a re-routable decap is preserved by the global VDD grid. Hence, they require little charge during the wake up process. This means turn-on time can be shortened and leakage energy consumed during wake up E_{ton} is reduced. By replacing parts of LDs with RDs, the energy overhead E_{LD} is decreased. Therefore, the total energy overhead of power gating E_{over} is significantly reduced. Compared with same amount of LDs or GDs, re-routable decaps occupy more on-chip area due to switches S_L and S_R.

Table 2.2: Comparison among Three Types of On-Chip Decaps

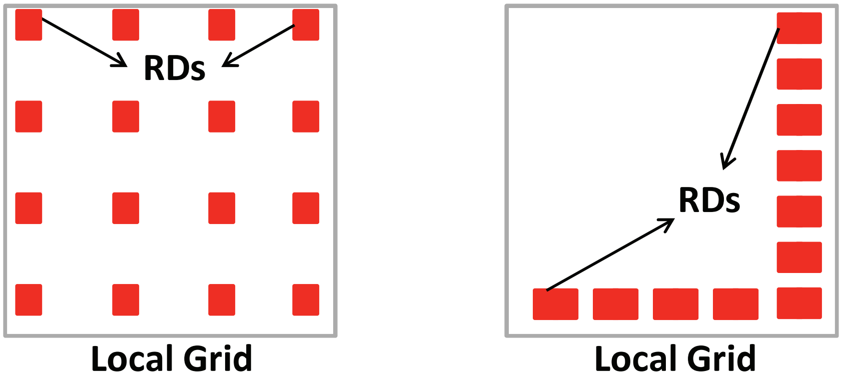
Type	Switching Noise Suppression	Rush Current Noise Suppression	Energy Overhead	Area Overhead
LDs	Excellent	Negative	$\surd$	–
GDs	Poor	Good	–	–
RDs	Good	Excellent	–	$\surd$

2.1.4.3 Design Strategy

The LD&GD&RD Strategy exploits re-routable decaps to reduce rush current noise and the energy overhead. Two design issues emerge with this strategy: allocation of re-routable decaps and the size of the S_L and S_R switches

Allocation of Re-Routable Decaps

Unlike typical on-chip decaps, re-routable decaps are reused by more than one local grids. On one hand, a re-routable decap acts as a local decap to suppress the switching noise of its own power domain. On the other hand, when the local grid is turned off, it acts as a global decap to suppress supply noises of other power grids. Hence, the allocation of re-routable decaps should consider both of these cases.

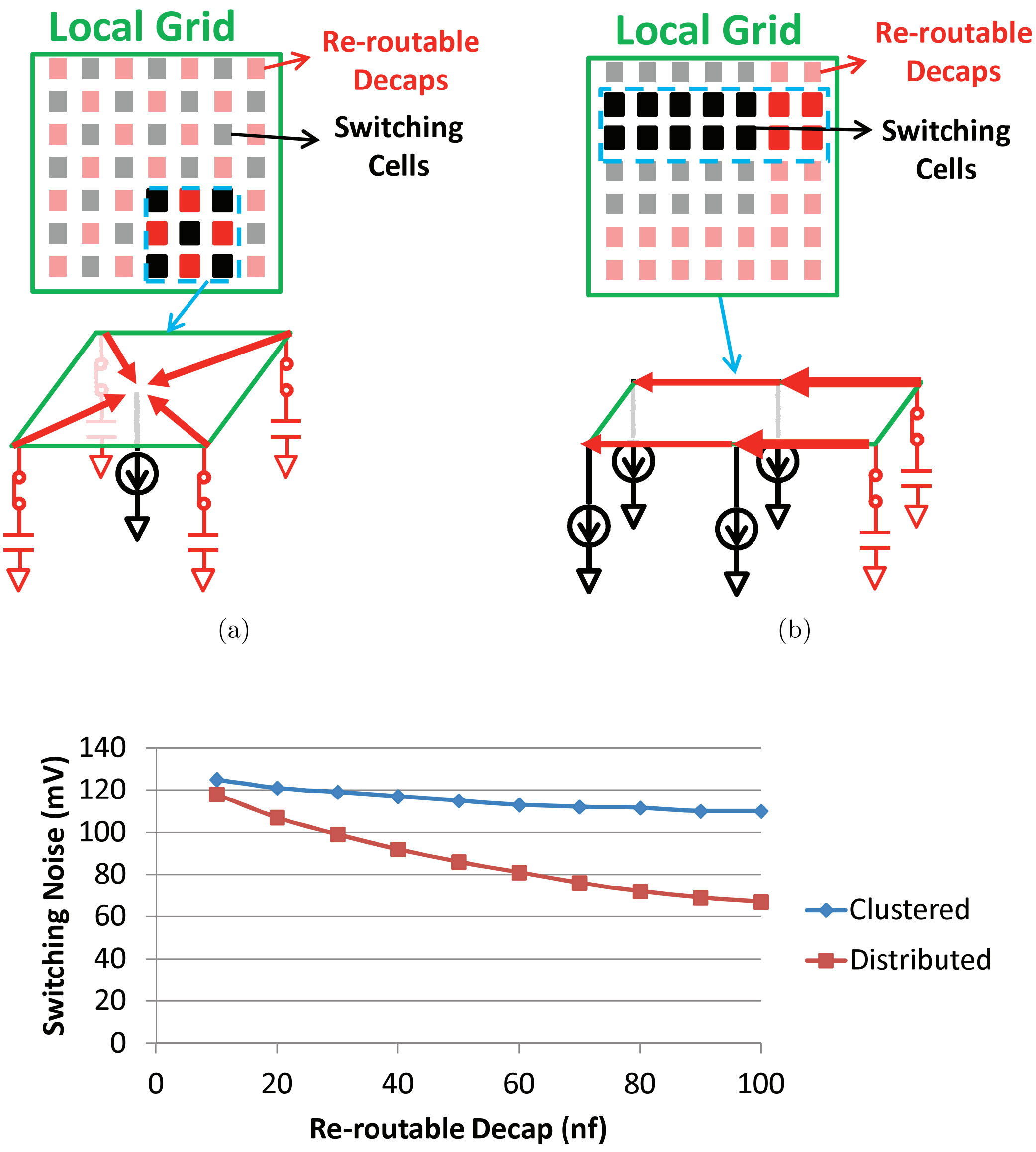


(a) (b)

Figure 2.16: Two different allocations of re-routable decaps: (a) distributed allocation; (b) clustered allocation.

Re-routable decaps can be allocated in two different ways. The first one is distributed allocation. As shown in Fig. 2.16(a), re-routable decaps are uniformly distributed on local grid A. The other one is clustered allocation that is shown in Fig. 2.16(b). Re-routable decaps are densely located at the boundaries of local grid A. The advantages and disadvantages of each allocation are discussed as follows.

Distributed allocation is advantageous to suppress switching noise. The resistance between a re-routable decap and a switching cell determines the efficiency of switching noise suppression. Through distributed allocation, re-routable decaps are



(a)

(b)

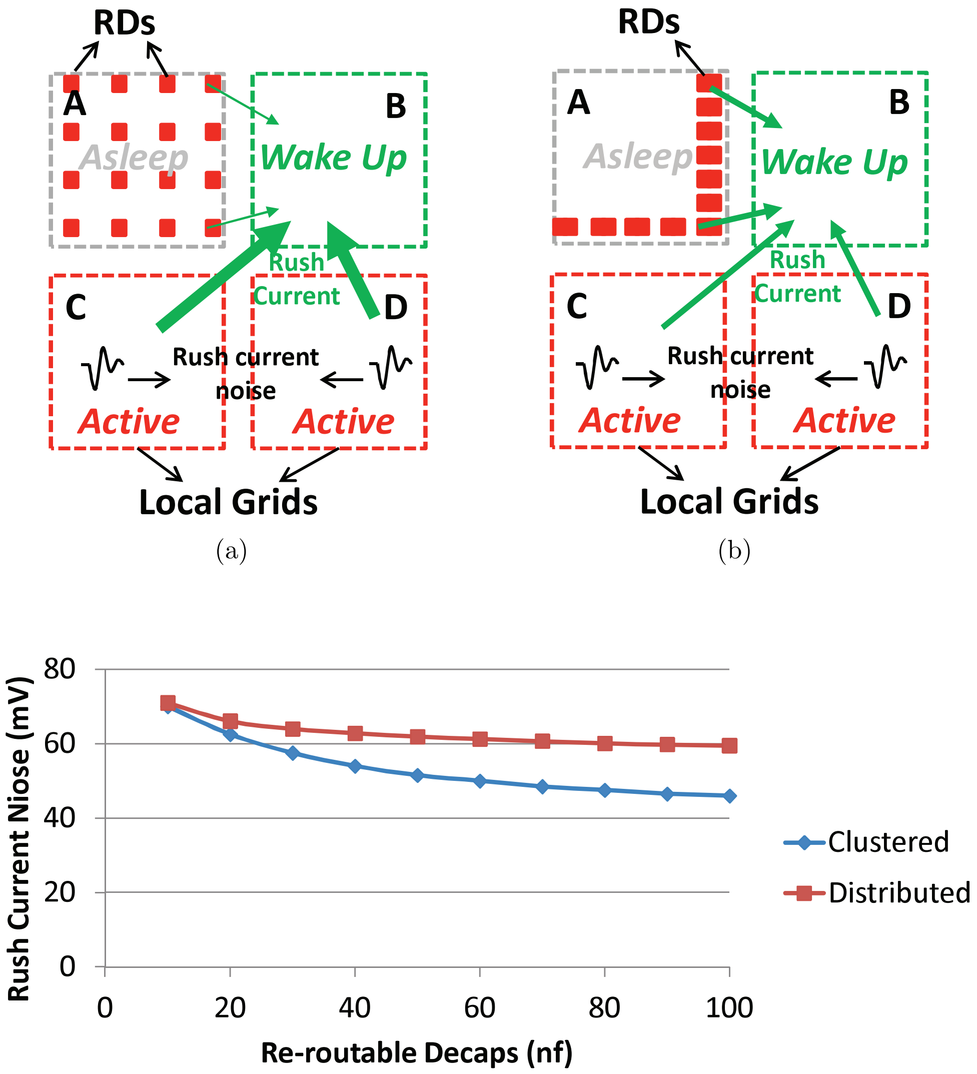
(c)

Figure 2.17: Switching noise suppression of re-routable decaps with different allocations. The simulations are based on the PDN model shown in Fig. 1.6. Only re-routable decaps are utilized in the circuit (no local decap or global decap). The amount of RDs is taken as a tuning parameter. The switching noises of the circuit with different amounts of RDs are monitored. The technology node is 45nm. (a) Distributed allocation of re-routable decaps. (b) Clustered allocation of re-routable decaps. (c) Switching noises with different re-routable decaps allocation.

located among the switching cells of local grid A as shown in Fig. 2.17(a). Hence, the switching noise of each switching cell is suppressed by the re-routable decaps nearby. In contrast, the re-routable decaps are located along the boundaries of local grid A in clustered allocation as shown in Fig. 2.17(b). Since they are allocated far away from most of the switching cells, large resistance weakens the suppression of switching noise. As shown in Fig. 2.17(c), the switching noise under distributed allocation is smaller than the one under clustered allocation with same amount of re-routable decaps. It indicates that RDs in distributed allocation are more efficient than the ones in clustered allocation to suppress switching noise.

On the other hand, clustered allocation has an advantage over distributed allocation to suppress rush current noise. When a local grid is turned off, its re-routable decaps are routed to the global VDD grid. As shown in Fig. 2.18(a) and 2.18(b), local grid A is turned off and the re-routable decaps of A act as global decaps to suppress the rush current noises of other active local grids (C and D). The noise is due to the rush current created by local grid B during its wake up process. For distributed allocation in Fig. 2.18(a), re-routable decaps are allocated far away from local grid B that is the source of rush current. Hence, such kind of allocation is disadvantageous to the suppression of rush current noise. In contrast, re-routable decaps are allocated along the boundaries of local grid A under clustered allocation. When the re-routable decaps are routed to the global grid, they are closer to local grid B than distributed allocation and thereby more current can be provided by these re-routable decaps. As shown in Fig. 2.18(c), with the same amount of re-routable decaps, clustered allocation is more efficient to suppress rush current noise.

As discussed above, a re-routable decaps is reused to suppress switching noise (own local grid) and to suppress rush current noise (other local grids) at different time. In order to enhance the efficiency, distributed allocation and clustered al-



(a)

(b)

(c)

Figure 2.18: Rush noise suppression of re-routable decaps with different allocations. The PDN model is shown in Fig. 1.6. Only local decaps and re-routable decaps are utilized in the circuit (no global decap). The amount of local decaps allocated in each local domain is 25nf. Re-routable decaps are only allocated in local grid A. The technology node is 45nm. (a) Distributed allocation of re-routable decaps on local grid A. (b) Clustered allocation of re-routable decaps on local grid A. (c) Rush current noises with different allocations.

location can be utilized together. We divide re-routable decaps into two groups. Re-routable decaps of the first group are uniformly distributed on the local grid to improve the efficiency of switching noise suppression. Re-routable decaps of the second group are allocated at local grid boundaries to improve the efficiency of rush current noise suppression. In order to determine the RD amount of each group, we propose a simulation based optimization flow that is discussed in Section 2.1.5.

For the special case discussed in Section 2.1.4.1, re-routable decaps are only used to preserve charge when the local grid is turned off. In this case, re-routable decaps are not used to suppress the rush current noises introduced by other local grids. Therefore, all the re-routable decaps can be allocated through distributed allocation.

Sizes of Switch S_L and S_R

Switch S_L connects a re-routable decap with a local grid. It determines the charge that can be provided by the re-routable decap for switching noise suppression. The size of S_L is constrained by two issues: area overhead and capacitance overhead. The area overhead is due to the addition of switch that is given by

$$A_0 = \frac{Area\ of\ S_L}{Area\ of\ decap}.$$

The capacitance overhead is another constraint of S_L. The series resistance of S_L reduces the efficiency of capacitance. Due to reduced efficiency, more capacitance is required to meet the power integrity requirement if we replace local decaps with re-routable decaps. The capacitance overhead is given by

$$C_0 = \frac{capacitance\ of\ RD}{equivalent\ capacitance\ of\ LD}.$$

Fig. 2.19 shows capacitance overhead and switch area overhead of the re-routable decaps required to reduce switching noise to 10% V_{DD}. As the width of S_L increases, the area overhead increases while the capacitance overhead decreases. In Section 2.1.5, we propose simulation based design optimization to determine the width of S_L in order to balance between capacitance overhead and area overhead.

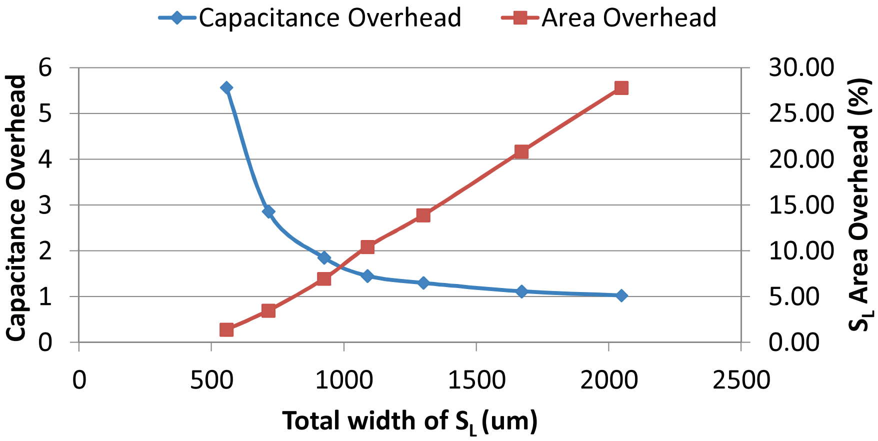


Figure 2.19: Capacitance overhead and switch area overhead of the re-routable decaps required to reduce switching noise to tolerable value. The maximum tolerable switching noise is 10% of V_{DD}. The circuit model is shown in Fig. 1.1. Only re-routable decaps are utilized in the circuit (no local decaps or global decaps). The re-routable decaps are allocated through distributed allocation. The technology node is 45nm.

Similar issues should be considered in the design of switch S_R. On one hand, the size of S_R should be large enough to suppress the rush current noise introduced by other local grids. On the other hand, the area overhead of the switch should be controlled to save limited on-chip white space.

2.1.5 Optimization Flow

In this section, we propose a simulation based optimization flow to design a power-gated PDN with single supply voltage automatically. Global decaps, local decaps, re-routable decaps and the turn-on time are taken as design parameters. Supply noises, leakage saving and area overhead are taken as components of the objective function. The simulation flow is proposed as shown in Fig. 2.20 to implement the LD&GD&RD Strategy that is discussed in Section 2.1.4.

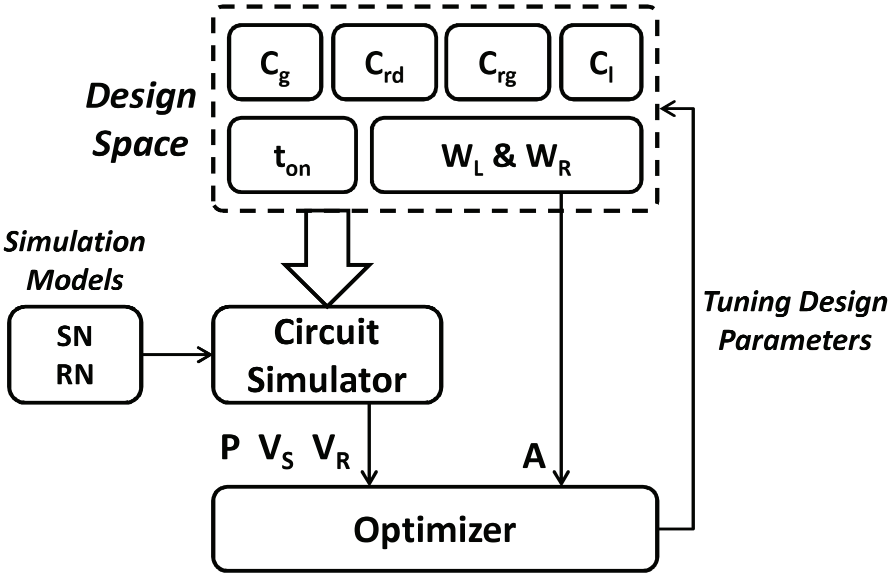


Figure 2.20: Simulation based optimization flow for PDN design with single supply voltage.

The design parameters of the strategy include the amount of LDs, GDs, and RDs in distributed allocation, RDs in clustered allocation, turn-on time, and total width of S_L and S_R. These design parameters are constrained as follows. The descriptions

Table 2.3: Design Parameters of Power-Gated PDN with Single Supply Voltage

V_S	maximum switching noise
V_R	maximum rush current noise
P	leakage power consumption
A	area overhead of the re-routable decaps' switches
C_l	amount of local decaps
C_g	amount of global decaps
C_{rd}	amount of re-routable decaps in distributed allocation
C_{rc}	amount of re-routable decaps in clustered allocation
C_{tot}	total on-chip decap budget
W_L	total width of switch S_L
W_R	total width of switch S_L
W_m	maximum width of re-routable decaps

of related parameters are listed in Tab. 2.3.

$$\begin{cases} C_g + C_l + C_{rd} + C_{rc} & \leq & C_{tot} \\ W_L + W_G & \leq & W_m \\ C_g, C_l, C_{rd}, C_{rc}, W_L, W_G & \geq & 0 \end{cases} \tag{2.8}$$

Two circuit models (SN and RN) are provided for the simulation. These two models share the same PDN structure. In model SN, all local grids are active. In model RN, only one local grid is active while the other local grids are asleep or waking up. Based on the design parameters selected from the design space and model SN, the maximum switching noise (V_S) can be obtained from the circuit simulation. The maximum rush current noise (V_R) and leakage consumption (P) can be obtained from the simulation of model RN. The area overhead (A) of re-routable decaps' switches is estimated based on the total width of S_L and S_G.

Based on V_S, V_R, P, and A, the optimizer evaluates the current design through an objective function and tune the parameters to improve the design for the next iteration. The objective function is given as

$$min \qquad f = f_s(V_s) + f_r(V_r) + f_p(P) + f_a(A), \tag{2.9}$$

where f_s and f_r are respectively the penalty functions of switching noise and rush current noise, f_p is the penalty function of leakage power consumption, and f_a is the penalty function of switch area overhead. The optimization flow is not restricted to any specific objective function but can use any generic function of V_s, V_r, P, and A. The formulation of each penalty function can be selected by designers.

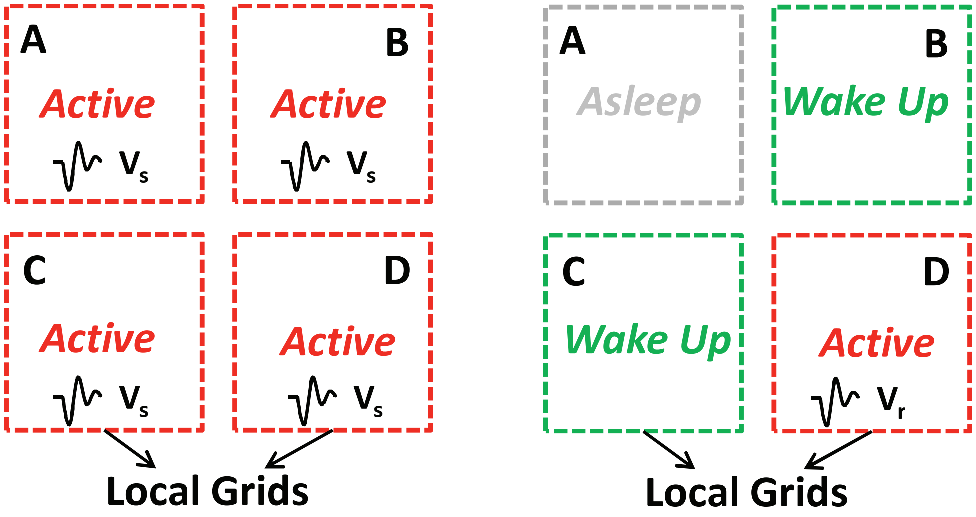
2.1.6 Experimental Results

In this section, we present the experimental results of the power-gated PDN design with single supply voltage.

The settings of the experiments are listed in Tab. 2.4. The interface of the optimizer and the optimization flow are implemented in C++. The package model parameters are from [11]. The power grids including four local grids are generated according to IBM power grid benchmarks [51].

Table 2.4: Experimental Setting of Power-Gated PDN with Single Supply Voltage

Single supply voltage	1V
Technology node	45nm
Average power	12W
On-chip Decap budget (C_{tot})	100nf
Maximum RD switch overhead(W_m)	1000μm
Maximum tolerable switching noise	9.5% of V_{DD}
Maximum tolerable rush current noise	0.5% of V_{DD}
Number of power domains	4
Size of PDN	120K Nodes
Circuit simulator	HSPICE C-2009-0.9
Optimizer	APPSPACK [52]



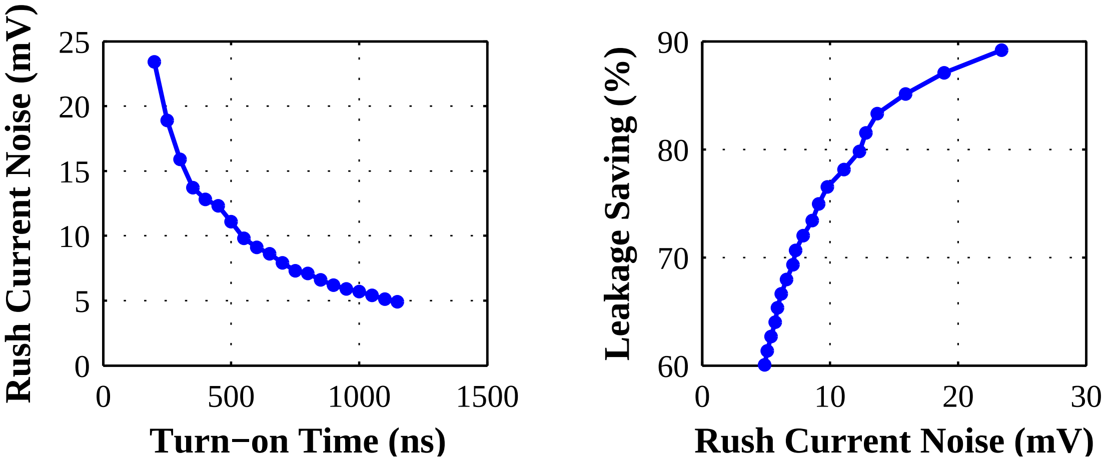
(a) (b)

Figure 2.21: Simulation models for optimization flow. (a) Model for switching noise simulation. (b) Model for rush current noise simulation.

The models used for simulation is shown in Fig. 2.21. The PDN structure includes 4 local grids. For the simulation of switching noise, all the local grids are active. For the simulation of rush current noise, local grid A is asleep, local grid D is active, local grids B and C are turning on.

We compare three different design strategies: LD only strategy, LD&GD strategy and LD&GD&RD strategy. For the LD only strategy, only local decaps are utilized in the PDN design. For the LD&GD strategy, both local decaps and global decaps are utilized. For the LD&GD&RD strategy, local decaps, global decaps and re-routable decaps are all used.

For the LD only strategy, rush current noise is mainly suppressed through extending the turn-on time. In Fig. 2.22(a), all designs meet the requirement of switching noise suppression (9.5% of V_{DD}). In order to reduce the rush current noise to 0.5% of V_{DD}, turn-on time has to be extended to 1000ns. Since turn-on time determines the opportunities of power gating, leakage saving is restricted by rush current noise. As shown in Fig. 2.22(b), rush current noise dramatically increases as more leakage is saved. In this figure, the leakage saving is normalized to the leakage power consumption without power gating. Therefore, the LD only strategy has limited leakage saving due to the tight interaction between rush current noise and leakage saving.

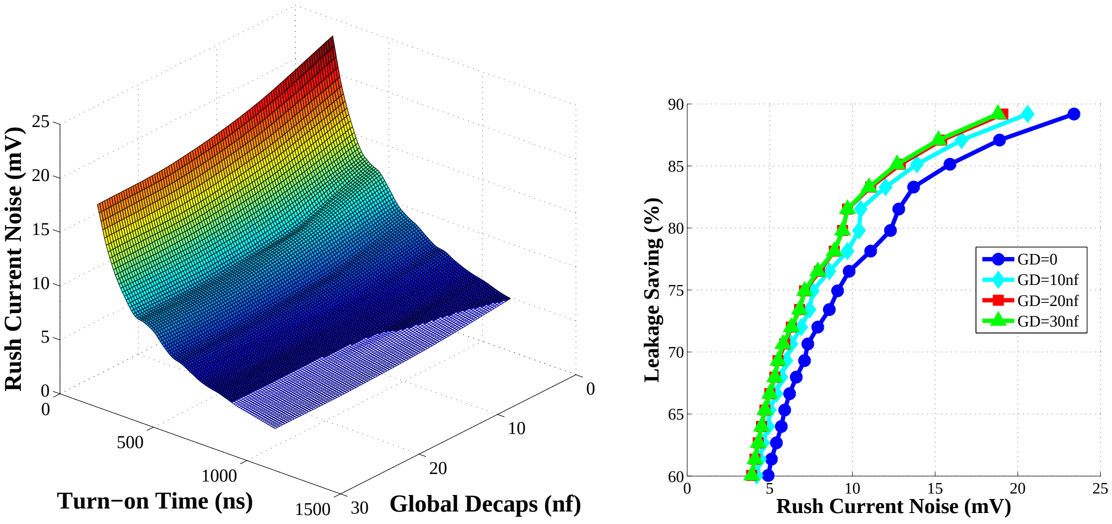


(a)

(b)

Figure 2.22: Rush current noise and leakage saving through the LD only strategy. Switching noise is reduced to 9.5% of V_{DD}. (a) Rush current suppression fully depends on extending turn-on time. (b) The interaction between leakage saving and rush current noise. Leakage saving is restricted by rush current noise. The leakage saving is normalized to the leakage power consumed without power gating.

For the LD&GD strategy, global decaps are used to suppress the rush current noise. Fig. 2.23(a) shows how rush current noise is influenced by turn-on time and the amount of global decaps. In this experiment, the switching noise is reduced to 9.5% of V_{DD}. The gray zone in the figure covers all feasible designs of which rush current noises are under 0.5% of V_{DD}. Compared with the LD only strategy, the feasible designs provided by the LD&GD strategy have shorter turn-on time. This is because the constraint of turn-on time is relaxed by global decaps. As shown in Fig. 2.23(b), the interaction between leakage saving and rush current noise is relaxed by global decaps. In other words, the LD&GD strategy can save more leakage power than the LD only strategy upon the same specification of supply noises.

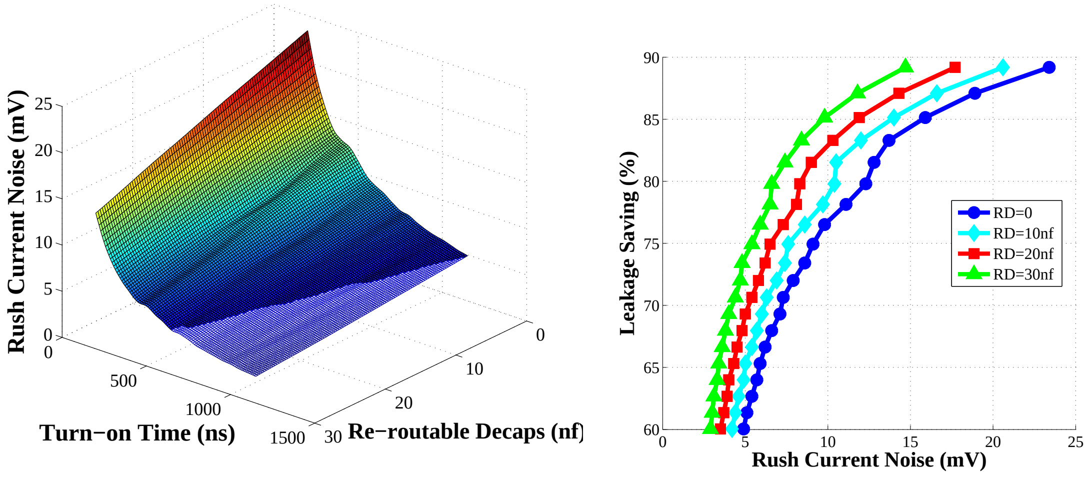


(a) (b)

Figure 2.23: Rush current noise and leakage saving through the LD&GD strategy. Switching noise is reduced to 9.5% of V_{DD}. (a) Rush current noise is suppressed by both turn-on time and global decaps. The gray zone in Fig. 2.23(a) covers the designs with rush current noise under 0.5% of V_{DD}. (b) Global decaps relax the interaction between leakage saving and rush current noise.

The LD&GD&RD Strategy exploits re-routable decaps to further reduce rush current noise. Fig. 2.24 shows rush current noise and leakage saving of the PDN designs under the LD&GD&RD strategy. In this experiment, only re-routable decaps and local decaps are used. Compared with the LD&GD strategy, the zone of feasible designs in Fig. 2.24(a) obviously extends. It indicates that re-routable decaps are more efficient to suppress rush current noise than the same amount of global decaps.

Fig. 2.24(b) shows that the interaction between leakage saving and rush current noise is further relaxed by the utilization of re-routable decaps.

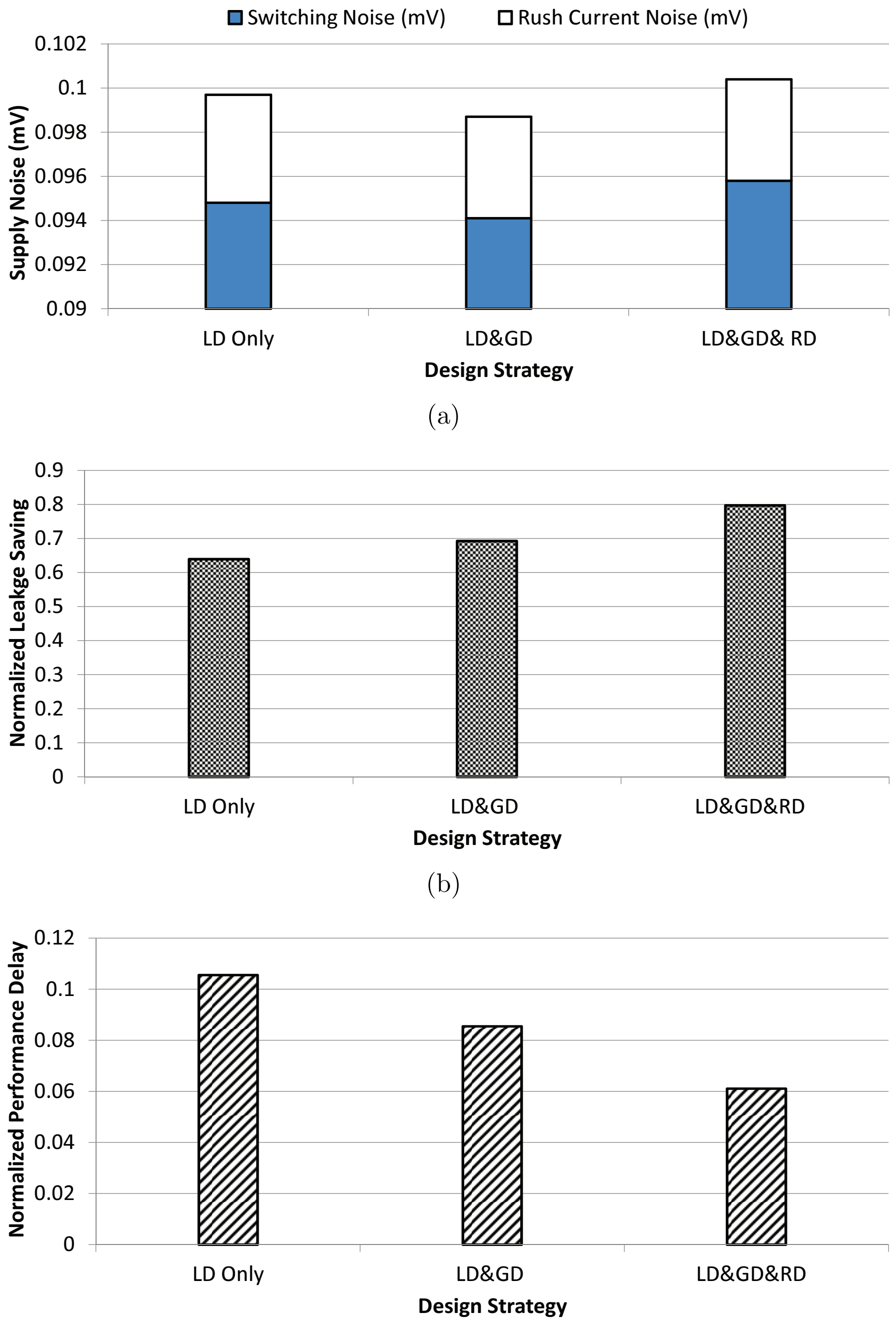


(a) (b)

Figure 2.24: Rush current noise and leakage saving through the LD&GD&RD strategy. No GD is used in order to evaluate the influence of re-routable decaps. Switching noise is reduced to 9.5% of V_{DD}. (a) Rush current noise is suppressed by both turn-on and re-routable decaps. The gray zone covers the designs whose rush current noises are under 0.5% of V_{DD}. (b) Re-routable decaps obviously relax the interaction between leakage saving and rush current noise.

Fig. 2.25(a) presents the optimized supply noises obtained from the LD only strategy, the LD&GD strategy and the LD&GD&RD strategy. Supply noises are important design concerns of a PDN design. The three strategies have similar performance of supply noises suppression. The maximum tolerable switching noise and rush current noise are respectively set as 9.5% and 0.5% as listed in Tab. 2.4. All the three strategies meet that specification of supply noises.

Fig. 2.25(b) and 2.25(c) respectively show the leakage saving and performance delay of optimization results obtained from the three strategies. The leakage saving



(a)

(b)

(c)

Figure 2.25: Comparison of optimization results obtained from the LD only strategy, the LD&GD strategy and the LD&GD&RD strategy. (a) Comparison of supply noises. (b) Comparison of normalized leakage savings. The leakage savings through different design strategies are normalized to the leakage consumption without power gating. (c) Comparison of normalized performance delays. The performance delays through different design strategies are normalized to the execution time without power gating.

is normalized to the total leakage consumption of the PDN design without power gating. The performance delay is normalized to the total execution time without power gating. The LD only strategy has no other means but extending the turn-on time to suppress the rush current noise. Turn-on time of the LD only strategy is extended long enough in order to meet the specification of rush current noise. Power gating with long turn-on time cannot be applied to short idle intervals that take up a large proportion of idle time. As a result, the normalized leakage saving of the LD only strategy achieves 60%. On the other hand, long turn-on time leads to a long delay of each power gating. Therefore, the performance delay is about 11% of the total execution time without power gating.

The LD&GD Strategy relaxes the interaction between rush current noise and turn-on time through the utilization of global decaps. Hence, the normalized leakage saving increases to 70% and the performance delay is reduced to 8.5%.

For the LD&GD&RD strategy, re-routable decaps are exploited to further reduce the rush current noise. Compared with global decaps, re-routable decaps are more efficient to suppress the rush current noise. In this case, the turn-on time can be significantly reduced. As a result, the tight interaction between the rush current noise and the leakage saving is relaxed by the re-routable decaps. In this case, this strategy saves about 80% leakage consumption that is the most leakage saving among the three strategies. The performance delay is reduced to 6.1%.

As shown in Tab. 2.5, the total decap budget (100nf) is not fully utilized in the design obtained though the LD only strategy. It is because that increasing local decap may lead to soaring rush current noise. Hence, the decap area only takes up 70% of the area of decap budget (100nf). However, this area saving is at the cost of leakage saving. The decap budget is fully used for both LD&GD strategy and LD&GD&RD strategy. This is because that they both have an effective mechanism

Table 2.5: Comparison among Three Design Strategies of Power-Gated PDN with Single Supply Voltage

Strategy	LD only	LD&GD	LD&GD&RD
Decap Budget(nf)	100	100	100
Local Decap(nf)	70	45	39
Global Decap(nf)	0	55	49
RD in distributed allocation (nf)	0	0	7
RD in clustered allocation (nf)	0	0	5
Switching Noise(mV)	95.3	97.1	97.3
Rush Current Noise(mV)	4.9	4.5	4.9
Total Supply Noise(mV)	100.2	101.6	102.2
Turn-on Time(ns)	1150	800	450
Leakage Saving[1]	60.0%	69.3%	79.8%
Performance Delay[2]	12.1%	8.5%	6.1%
Decap Area[3]	70%	100%	103%

[1] normalized to the leakage power consumed without power gating;
[2] normalized to the execution time without power gating;
[3] normalized to the area of total decap budget (100nf).

(GDs or RDs) to suppress rush current noise. Compared with the LD&GD strategy, the LD&GD&RD strategy consumes 3% more area. This area overhead is due to the switches of re-routable decaps. More details about these three design strategies are listed in Tab. 2.5.

2.1.7 Summary

In this section, two decoupling strategies are proposed to address the interaction between power integrity and power efficiency. Compared with existing power-gated PDN design works, we utilize global decaps and re-routable decaps to suppress rush current noise. These decoupling strategies relax the interaction between turn-on time and rush current noise. Hence, more leakage can be saved through shortening the turn-on time. In addition, our proposed strategies provide methods to balance between switching noise and rush current noise. A simulation-based optimization flow is proposed to design PDNs with proposed strategies. The experimental results have shown that leakage saving is increased by 30% based upon the proposed methodology compared with conventional PDN design with single supply voltage.

2.2 Design Power-Gated PDNs with Multiple Supply Voltages

As discussed in the last section, the trade-offs between power integrity and power efficiency exit in power-gated PDN designs. These trade-offs vary with the supply voltage significantly. Hence, it is more difficult to meet the design concerns of power-gated PDNs with multiple supply voltages. In this section, we take the a power-gated PDN design with two supply voltages as an example to discuss the challenge and propose a flexible decoupling strategy.

2.2.1 Background

As CMOS technology scales down, dynamic power consumption of VLSI circuits becomes a significant challenge. More and more systems are operated according to tasks' workloads or priorities. For example, a processor is supposed to operate at highest frequency to process critical tasks while it may slow down to process other non-critical tasks. In this case, the dynamic power consumption (CV_{dd}^2f) can be significantly saved though linearly reduction in the supply voltage and operating frequency [53]. Dynamic voltage scaling (DVS) or dynamic voltage and frequency scaling (DVFS) are widely applied to modern processors to provide different operating points. Multiple supply voltages are required to implement these power management techniques. Power gating can be combined with DVS or DVFS to further reduce the static power dissipation at each operating point [7, 54].

However, the designs of power-gated PDNs with multiple supply voltage are very complex. First, as discussed in the last section, trade-offs among switching noise, rush current noise, and energy saving exit at each supply voltage level. The power integrity and power efficiency requirements must be met at each operating point. Second, the design trade-offs vary with the supply voltage. For higher V_{DD}, power integrity is harder to achieve. Hence, power integrity is more important at higher voltage level. For lower V_{DD}, the break even time of power gating becomes longer and thereby energy saving is the dominant design concern. Most of existing works rarely consider the trade-off variation problem. The typical design solution is to design the PDN for the worst case. For example, the supply noises increase with the supply voltage. Hence, the decaps are allocated to meet power integrity requirement at higher V_{DD} level. However, this configuration is pessimistic for the power integrity condition at lower V_{DD}.

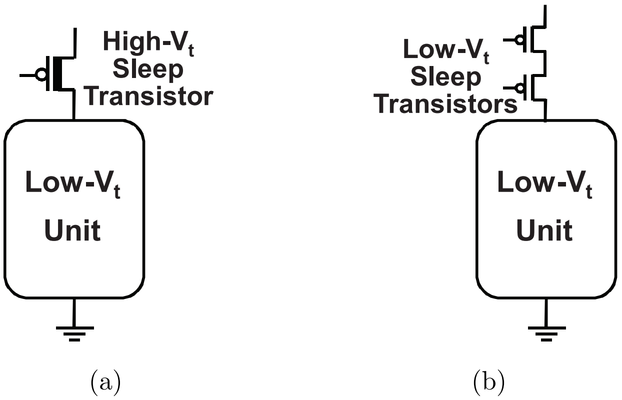


Figure 2.26: Structures of sleep transistors. (a) High-threshold sleep transistor. (b) Stacked low threshold voltage sleep transistor.

In order to address these problems, we analyze the power-gated PDNs with multiple supply voltages in this section. The PDN model used in this section is a little different to the one discussed in the last section. First, the DC voltage source can provide two voltage values (1V and 0.6V). Second, the structure of sleep transistor is changed. The power-gated power delivery networks (PDNs) with supply voltage higher than 1V are usually implemented by multi-threshold CMOS (MTCOMS) [55, 14]. As shown in Fig. 2.26(a), the logic cells are implemented by low-threshold CMOS to reduce delay and sleep transistor is implemented by high-threshold CMOS to reduce leakage. However, MTCOMS structure cannot be applied to sub-1V V_{DD} condition duo to the high IR droop and long wake-up time. Two series connected low-V_t transistors are used to implement power gating with low supply voltage, as shown in Fig. 2.26(b). The sub-threshold leakage is reduced by transistors' stack effect of this structure.

2.2.2 Design Concerns and Trade-Offs

As discussed in Sections 2.1.3 and 2.1.4, the decap configuration (local/global/re-routable decaps) at certain supply voltage level is determined based on leakage saving and power integrity. However, the trade-off between these two design concerns varies with the supply voltage that increases design complexity of the power-gated PDN with multiple supply voltages.

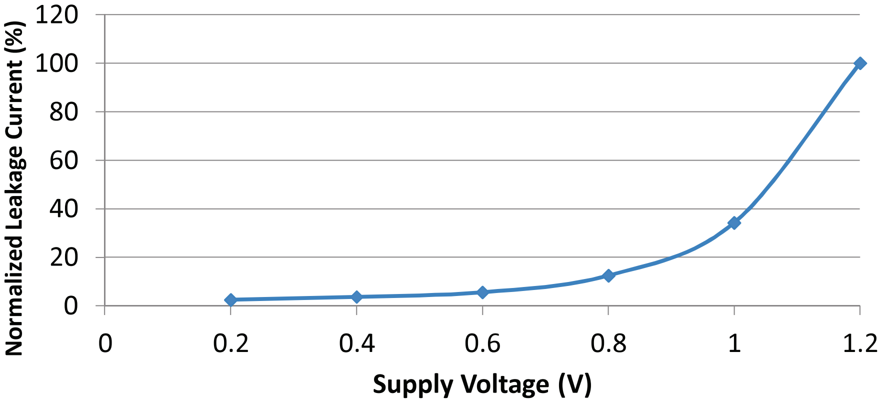


Figure 2.27: Normalized leakage current of an inverter increases with the supply voltage (V_{DD}). Leakage current is normalized to the value when V_{DD}=1.2V. The technology node is 45nm.

Power gating is exploited to reduce leakage power consumption that includes the sub-threshold leakage and gate tunneling leakage. As the process technology downscales into the deep nanometer range, high-κ dielectric is widely applied to MOSFET to reduce the gate leakage. Hence, sub-threshold leakage becomes the dominant as the process technology scales down. The sub-threshold current has an exponential relationship with threshold voltage (V_{TH}) that is approximately given by

$$I_{leak} \propto exp(\frac{V_{gs} - V_{TH}}{nv_T}) \tag{2.10}$$

where V_{TH} is the threshold voltage, v_T is the thermal voltage. A reduction of V_{TH} occurs at higher drain-source bias (V_{ds}) due to drain induced barrier lowering (DIBL) which is presented as

$$V_{TH} = V_{TH0} - \eta V_{ds}, \tag{2.11}$$

here V_{TH0} is the threshold voltage when V_{ds}=0, and η is the coefficient of DIBL. As a result, when the supply voltage of logic devices decreases linearly, the leakage current I_{leak} is reduced exponentially [56]. Fig. 2.27 shows the normalized leakage current of an inverter at different supply voltages. When the supply voltage decreases from 1.2V to 0.6V, the leakage current is reduced by about 20 times. As mentioned in Section 2.1.1, break even time (T_{BE}) of power gating is the time during which leakage saving compensates energy overhead. Since the energy overhead is mainly used to recharge the capacitance of the local grids, we have

$$E_{over} \propto C_{local} V_{DD}^2, \tag{2.12}$$

where C_{local} is the equivalent local decaps that include no-switching logic cells that act as capacitors and decoupling capacitance cells. The break even time can be estimated as

$$T_{BE} = \frac{E_{over}}{P_{leak}} \propto \frac{V_{DD}}{I_{leak}}, \tag{2.13}$$

where P_{leak} is the leakage power and I_{leak} is the average leakage current that exponentially decreases with V_{DD}. Hence, the break even time increases as the supply voltage decreases. It means that leakage consumption is harder to be saved through power gating at lower V_{DD}.

The switching current created by switching cells and the rush current created during wake-up process both superlinearly increase with V_{DD}. As a result, the ratio of switching noise or rush current noise to V_{DD} increases as supply voltage linearly increases. In other words, it is harder to meet the power integrity specifications at higher V_{DD}.

Table 2.6: Design Trade-Off Variations with Supply Voltage

V_{DD}	Switching Noise/V_{DD}	Rush Current Noise/V_{DD}	T_{BE}
High	↑	↑	↓
Low	↓	↓	↑

↑increase ↓decrease

The trade-offs with different supply voltages are summarized in Tab.2.6. It indicates that the design trade-offs change as the system switches between different voltage operating points. Power integrity is the dominant design concern at high-V_{DD} while leakage saving is the critical design concern at low-V_{DD}.

2.2.3 Proposed Diversity Decap Strategy

For typical PDN designs, only local decaps are used (LD only strategy). In this case, the required amount of local decaps varies with the supply voltage. As shown in Fig. 2.28, the required local decaps decrease as V_{DD} scales down when the total supply noise tolerance is kept as a fixed percentage of the nominal V_{DD}. In order to meet the power integrity requirement in the worst case, local decaps are designed to suppress the switching noise at highest V_{DD}. However, such amount of local decaps is superfluous at lower V_{DD}. Superfluous local decaps create extra rush current noise and thereby limit turn-on time shrinking. As a result, power gating has fewer opportunities to save leakage at low V_{DD}.

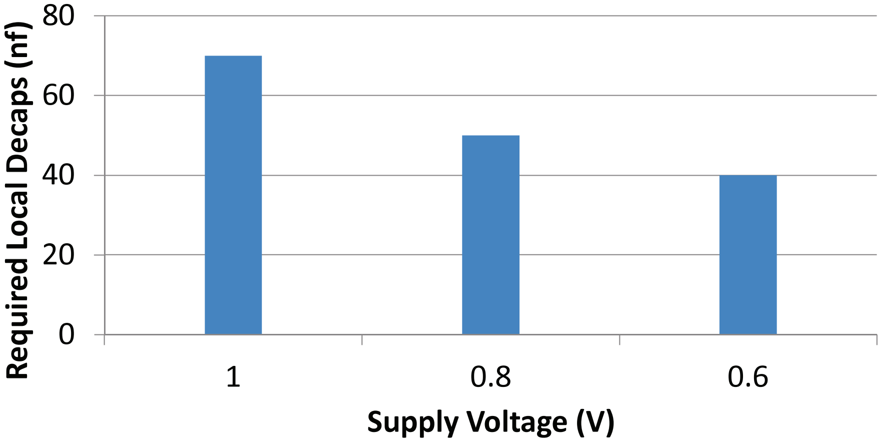


Figure 2.28: The decaps required at different supply voltages for the LD Only Strategy. The maximum tolerable supply noise is 10% of V_{DD}. The technology node is 45nm.

Obviously, a fixed decap configuration cannot adapt to the design trade-off changes with V_{DD}. A flexible decoupling strategy is proposed here using RDs. Two types of RDs are used: (a) regular RDs illustrated in 2.29(a) and (b) global RDs illustrated in 2.29(b). When the local grid is active, regular RDs are connected to the local grid and global RDs are connected to the global VDD grid. When the local grid is idle, both regular RDs and global RDs are connected to the global VDD grid. Regular RDs make sure that the design has enough decaps to suppress switching noise. Global RDs are used to further reduce rush current noise and thereby increase leakage saving. A flexible decap configuration is provided through tuning the proportion between these two types of RDs. At high V_{DD}, all RDs are used as regular RDs since it is the worst case for switching noise. As V_{DD} decreases, leakage saving becomes the main design concern. Hence, the proportion of global RDs is increased to further reduce rush current noise. As a result, the optimal design can be implemented at each V_{DD} level through diverse decap configurations.

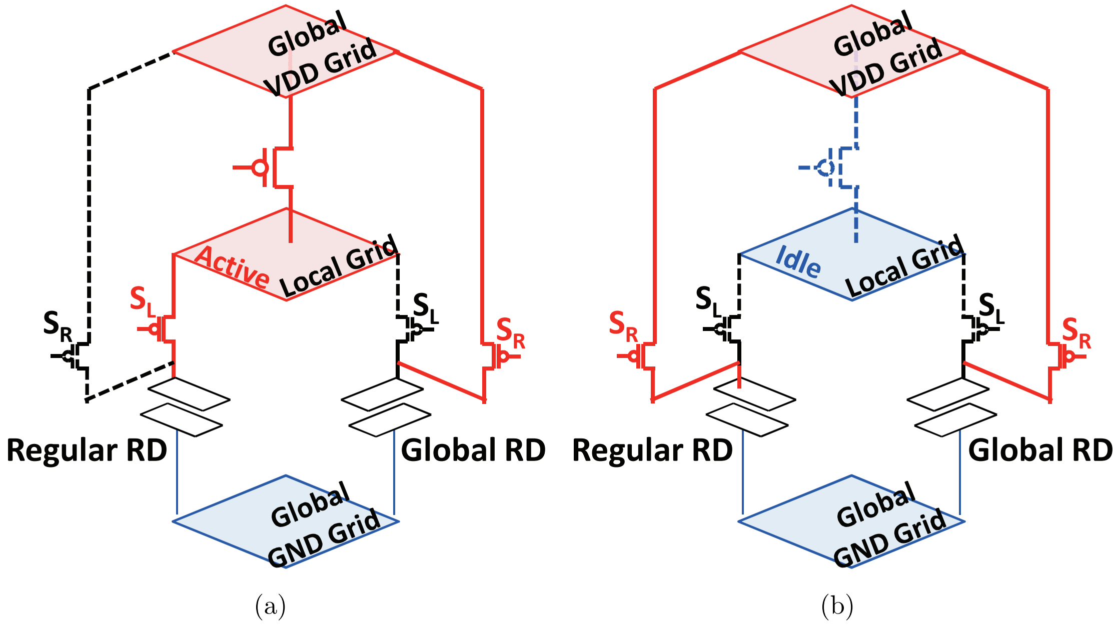


Figure 2.29: Usages of regular RD and global RD. (a) When the local grid is active, regular RD is connected to the local grid and global RD is connected to the global VDD grid. (b) When the local grid is idle, both regular RD and global RD are connected to the global VDD grid.

2.2.4 Optimization Flow

For a PDN design with multiple supply voltages, we only consider the special case where RDs are used to suppress the rush current noise created by their own local grid. Since the RDs are not used as global decaps, all of them are allocated in distributed allocation.

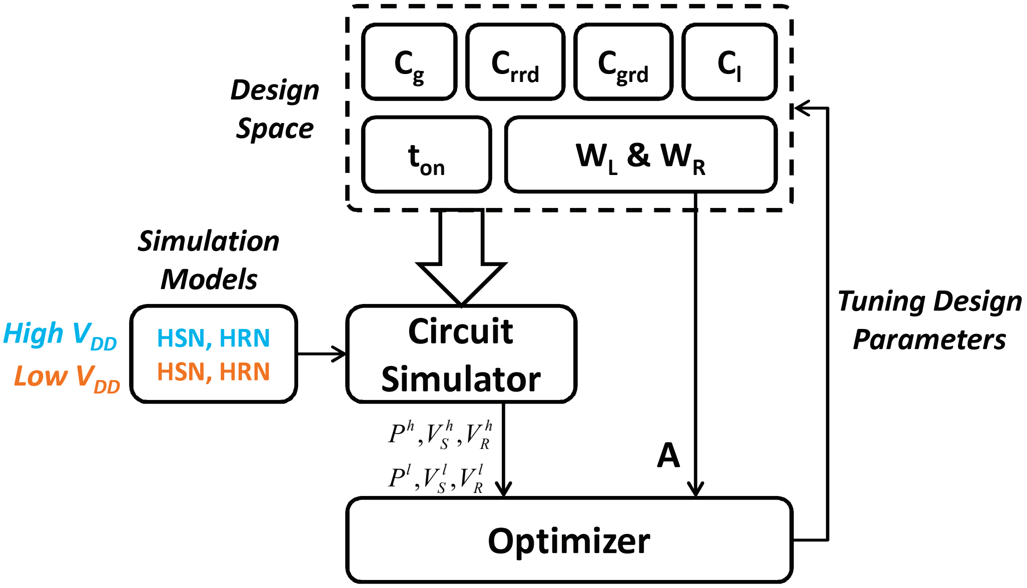


Figure 2.30: Simulation based optimization flow with two supply voltages.

The optimization flow for a PDN design with two supply voltages (V_{DD}^h and V_{DD}^l) is shown Fig. 2.30 which can be extended to handle a larger number of supply levels.V_{DD}^h and V_{DD}^l respectively indicate the high and low supply voltage.

The design parameters referred include the amount of LDs, GDs, regular RDs, global RDs, turn-on time, and total width of S_L and S_R. The descriptions of these parameters are listed in Tab. 2.7. The constraints of the parameters are given as follows.

Table 2.7: Design Parameters of Power-Gated PDN with Two Supply Voltages

$V_S^{h(l)}$	switching noise with $V_{DD}^{h(l)}$
$V_R^{h(l)}$	rush current noise with $V_{DD}^{h(l)}$
$P^{h(l)}$	leakage power consumption $V_{DD}^{h(l)}$
A	area overhead of the RDs' switches
C_l	amount of local decaps
C_g	amount of global decaps
C_{rrd}	amount of regular re-routable decaps
C_{grd}	amount of global re-routable decaps
C_{tot}	total on-chip decap budget
W_L	total width of switch S_L
W_R	total width of switch S_L
W_m	maximum width of re-routable decaps

$$\begin{cases} C_g + C_l + C_{rrd} + C_{grd} & \leq \quad C_{tot} \\ W_L + W_G & \leq \quad W_m \\ C_g, C_l, C_{rrd}, C_{grd}, W_L, W_G & \geq \quad 0 \end{cases} \tag{2.14}$$

Four simulation models are used for the PDN design with two supply voltages. HSN and LSN are respectively for the switching noise simulations at V_{DD}^h and V_{DD}^l. HRN and LRN are respectively for the rush current noise simulations at V_{DD}^h and V_{DD}^l. Based on the design parameters and simulation models, the maximum switching noise and rush current noise, the leakage consumption at V_{DD}^h and V_{DD}^l are obtained from circuit simulations.

Based on the outputs of circuit simulations, the optimizer evaluates the current design through an objective function and tune the parameters to improve the design for the next iteration. The objective function is given as

$$min \qquad f = f_s^h + f_s^l + f_r^h + f_r^l + f_p^h + f_p^l + f_a, \tag{2.15}$$

where $f_s^{h(l)}$ is the penalty function of switching noise at $V_{DD}^{h(l)}$, $f_r^{h(l)}$ is the penalty function of rush current noise at $V_{DD}^h(l)$, $f_p^{h(l)}$ is the penalty function of leakage power consumption at $V_{DD}^{h(l)}$, and f_a is the penalty function of switch area overhead. The detailed formulation of each penalty function can be selected by designers. The parameters referred in the flow are described in Tab. 2.7.

2.2.5 Experimental Results

In this section, we present the experimental results of the power-gated PDN design with multiple supply voltages.

The settings of the experiments are listed in Tab. 2.8. The interface of the optimizer and the optimization flow are implemented in C++. The package model and power grids are same as the PDN with single supply voltage in Section 2.1

Table 2.8: Experimental Setting of Power-Gated PDN with Two Supply Voltages

High supply voltage (V_{DD}^h)	1V
Low supply voltage (V_{DD}^l)	0.6V
Technology node	45nm
Average power	12W
On-chip Decap budget (C_{tot})	100nf
Maximum RD switch overhead(W_m)	1000μm
Maximum tolerable switching noise	9.5% of V_{DD}
Maximum tolerable rush current noise	0.5% of V_{DD}
Number of power domains	4
Size of PDN	120K Nodes
Circuit simulator	HSPICE C-2009-0.9
Optimizer	APPSPACK [52]

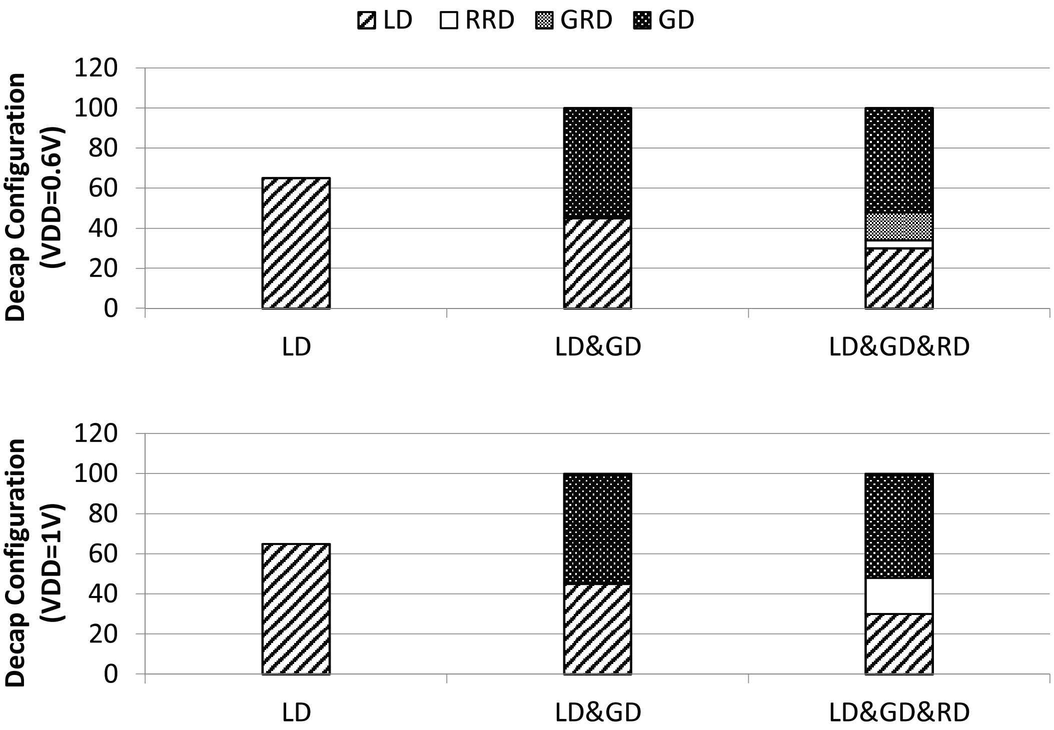


Figure 2.31: Decap configurations with two supply voltages. The total decap budget is 100 nf.

Through the optimization flow proposed in Section 2.2.4, the optimal decap configurations of the three strategies are obtained as shown in Fig. 2.31. For the LD only or the LD&GD strategies, the decap configuration is fixed at the two supply voltages. The LD&GD&RD strategy provides flexible decap configurations for two supply voltages. The total re-routable decaps in the design is 18nf. At the low voltage level (V_{DD}=0.6V), these re-routable decaps work as 4nf regular RDs and 14nf global RDs. Regular RDs act as local decaps when the local grid is active and act as global decaps when the local grid is idle. Global RDs are connected to the global VDD grid no matter the local grid is active or idle. At the high voltage level ($V_{DD} = 1V$), all re-routable decaps are used as regular RDs to enhance the suppression of supply noises.

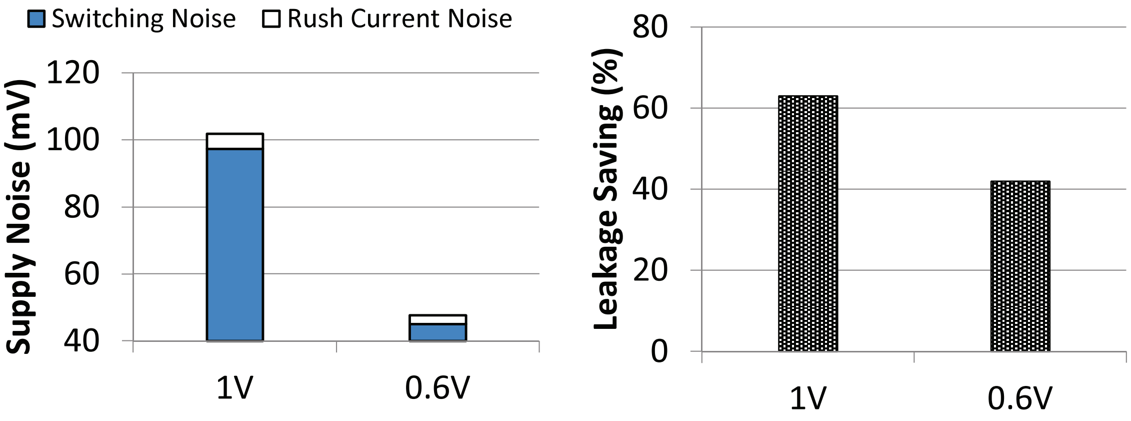


Figure 2.32: Supply noises and leakage saving of the LD only strategy.

Supply noises and leakage saving of the LD only strategy are shown in Fig. 2.32. For the LD only strategy, the amount of local decaps is determined by the switching noise at high V_{DD}. Hence, the supply noises meet the power integrity specification at $V_{DD} = 1V$. On the other hand, the total supply noise is much smaller than the maximum tolerable voltage drop ($10\%V_{DD}$) at $V_{DD} = 0.6V$. Although this result is advantageous to power integrity, it indicates that parts of the local decaps are unnecessary. These unnecessary local decaps increase rush current noise that impairs the leakage saving. As a result, the power gating at low V_{DD} only saves 40% of leakage consumption.

Supply noises and the leakage saving of the LD&GD strategy are shown in Fig. 2.33. This strategy is similar to the LD only strategy of which the decap configuration is fixed. As a result, the leakage saving at low V_{DD} is still limited by a large amount of local decaps that is unnecessary at low voltage level.

The LD&GD&RD Strategy provides different decap configurations for two supply voltages. At the high voltage level, all the re-routable decaps are used as regular RDs to make sure that supply noises meet the power integrity specification. As V_{DD}

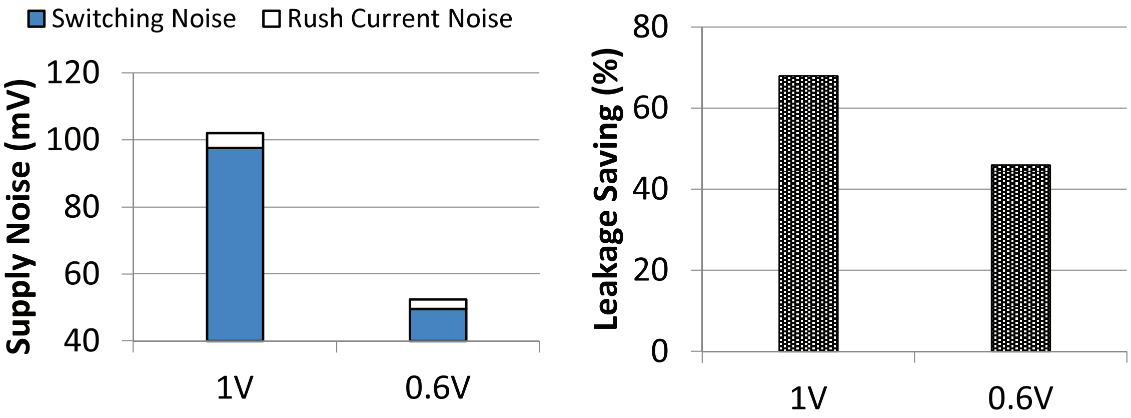


Figure 2.33: Supply noises and leakage saving of the LD&GD strategy.

decreases, parts of re-routable decaps are used as global RDs to suppress the rush current noise. As a result, turn-on time is further shortened and thereby the leakage saving increases to 70%.

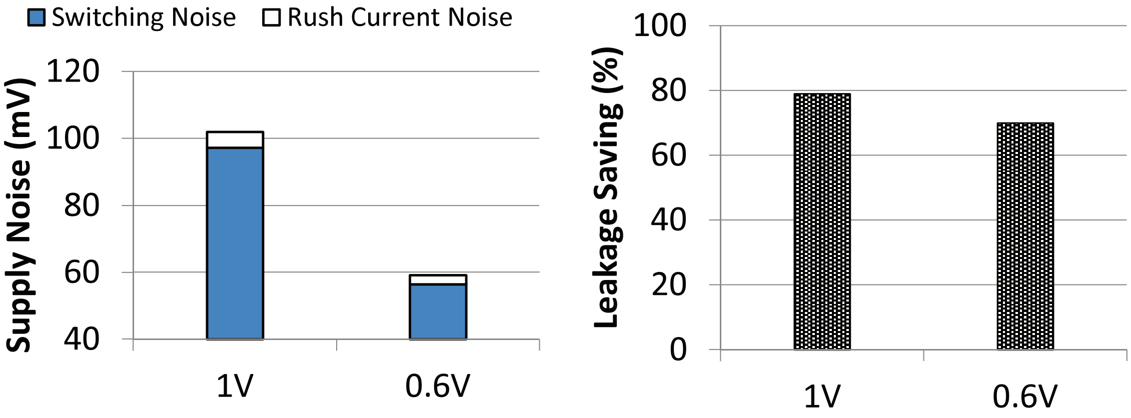


Figure 2.34: Supply noises and the leakage saving of the LD&GD&RD strategy.

As shown in Tab. 2.9, the decap area of LD only strategy takes up 65% of the area of decap budget (100nf). This is because that local decaps may increase the rush current noise. The decap budget is fully used for both LD&GD strategy and

LD&GD&RD strategy. Compared with the LD&GD strategy, the LD&GD&RD strategy consumes 5% more area due to the switches of re-routable decaps. More details are presented in Tab. 2.9.

Table 2.9: Comparison Among Three Design Strategies of Power-Gated PDN with Two Supply Voltages

Strategy	LD only		LD&GD		LD&GD&RD	
	0.6V	1.0V	0.6V	1.0V	0.6V	1.0V
L. Decaps(nf)	65	65	45	45	30	30
G. Decaps(nf)	0	0	55	55	52	52
Regular RDs(nf)	0	0	0	0	4	18
Global RDs(nf)	0	0	0	0	14	0
S. Noise(mV)	45.0	97.3	49.6	97.6	56.4	97.2
R. Noise(mV)	2.7	4.5	2.8	4.4	2.7	4.7
Tot. Noise(mV)	47.7	101.8	52.4	102	59.1	101.9
t_{on}(ns)	850	1000	700	800	400	450
Leak. Saving[1]	42%	63%	46%	68%	70%	79%
Decap Area[2]	65%		100%		105%	

[1] normalized to the leakage power consumed without power gating;

[2] normalized to the area of total decap budget (100nf).

2.2.6 Summary

In this section, a decoupling strategy is proposed for power-gated PDN designs with two supply voltages. LD&GD&RD strategy provides flexible decap configurations for different supply voltages. For higher supply voltage, all the re-routable decaps act as local decaps to suppress the switching noise when the local grid is active. For lower supply voltage, parts of the re-routable decaps act as global decaps to further suppress rush current noise. A simulation-based optimization flow is proposed to design the PDNs with proposed strategies. The experimental results show that our proposed flexible decap strategy achieves the optimal performance at both voltage levels.

3. SYSTEM/CIRCUIT CO-OPTIMIZATION STRATEGIES FOR DVFS

Fig. 1.4 shows the structure of DVFS design. A DVFS design is compose of the circuit-level design and system-level design. DC-DC converter is the main design on circuit-level. It is used to provide different supply voltages for the operating points. DVFS controller is the main design on system-level. A DVFS controller evaluates operating points and select the optimal one for the following operative period. For most of DVFS designs, the DC-DC converter and the DVFS controller are designed separately. However, the circuit-level and system-level designs highly interact with each other. The cross-layer design trade-offs significantly influence the performance of entire DVFS system.

In this paper, we proceed by first analyzing design trade-offs at the circuit level and the system level respectively. Then, the interaction between the DC-DC converter design and the DVFS controller is studied. As an intermediate study, we show that performing system-level policy optimization without considering circuit-level design can lead to suboptimal power and performance trade-offs. Finally, we demonstrate the benefit of cross-layer co-optimization of online-learning based DVFS controller and DC-DC converter and develop a practical design flow. The proposed design strategy is evaluated based on single-core processors, dual-core processors with global DVFS, and power-gated processors with DVFS respectively.

3.1 Design of DVFS for Single-Core Processors

Single-core processor has simple architecture and the DVFS for single-core processor has typical design trade-offs. In this section, we take DVFS of single-core processor as the start point to discuss the design issues of DVFS and propose the co-optimization design flow.

3.1.1 Background

The structure of DVFS system is presented in Fig. 1.4. A processor with DVFS can operate at different operating points. Each operating point is pair of supply voltage and operating frequency. The working set is composed of all the operating points of the processor. The DC-DC converter and the DVFS controller are respectively the circuit-level and system-level components of DVFS system. The DVFS technique is launched periodically as follows.

At the system level, the DVFS controller evaluates all operating points in the working set at the end of each operative period. The evaluation includes two aspects: the processor energy consumption and the performance delay when the processor operates at given operating point. It is hard and expensive to obtain the real-time information of energy or delay. In most of works, the energy consumption and perforce delay are estimated by CPU statistic data, such as cache misses, CPU usage, and instructions per cycle. These data can be easily monitored and accessed through using CPU performance counters. Hence, operating point evaluation is based on an evaluation model that is a mapping between statistic data to expected operating points. It means the energy consumption and performance delay is balanced when the processor operates at the expected operating point. An operating point is evaluated through comparing with the expected operating point. The evaluation results of current and previous operative periods are utilized to generate an comprehensive score for each operating point though certain algorithm. The DVFS controller selects the optimal operating point according to their sores.

At the circuit level, the DC-DC converter transform output voltage to the selected operating point. DC-DC converter output voltage and processor operating frequency scaling procedure is shown in Fig. 3.1. This DVFS procedure is commonly used in

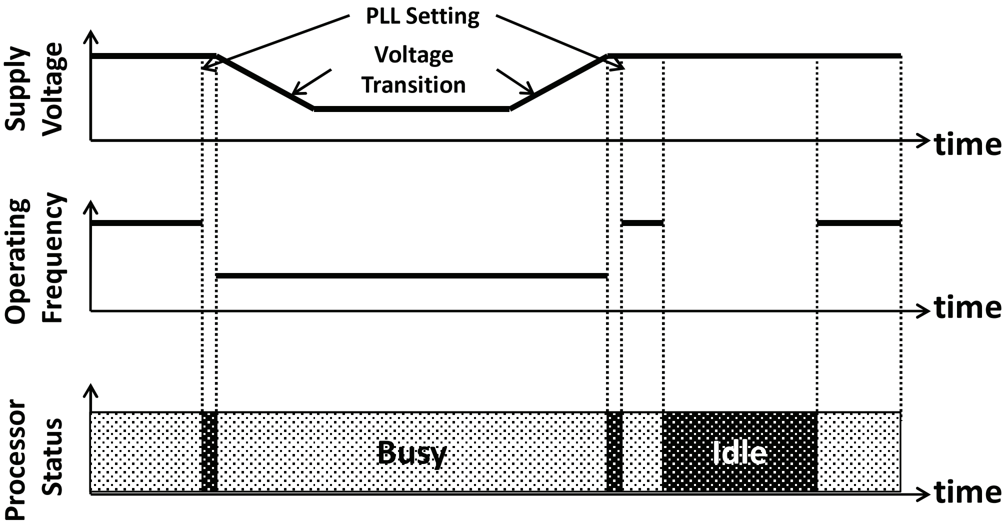
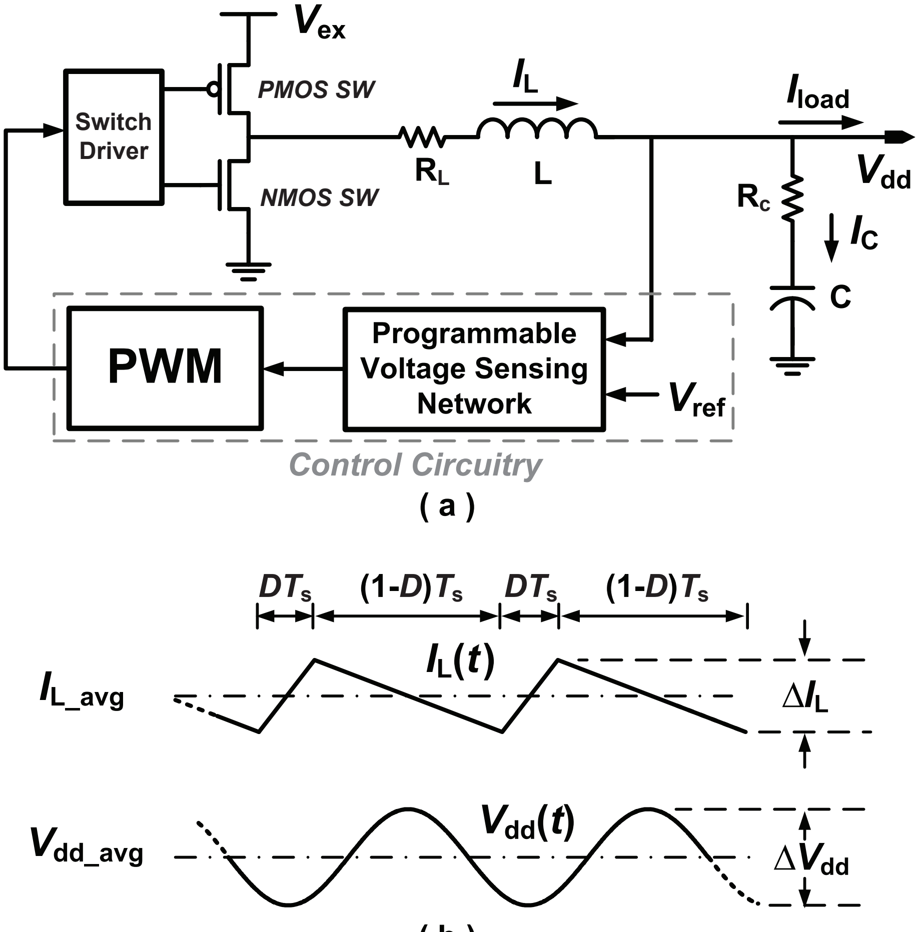


Figure 3.1: Supply voltage and operating frequency scaling procedure of single-core processor.

modern microprocessors [20, 7]. For the down-scaling of the clock frequency, the PLL setting is initialized firstly. The CPU is halted during the PLL locking time. The change of the PLL setting can be finished in several cycles by the technique proposed in [57]. The frequency is settled down to the selected value after the PLL is settled. Then the supply voltage decreases gradually to the value of the new operating point. The opposite procedure happens when an operating point of higher performance is employed.

3.1.2 Circuit-Level Design

Fig. 3.2(a) illustrates the standard PWM-based DC-DC converter we employed in our study. In principle, the design of a transistor-level DC-DC converter can be very complex. Without loss of generality, this discussion of converter design employs a behavioral model of the converter to capture the key design aspects that influence the succeeding power delivery system [58]. The average output voltage (V_{dd_avg}) as well as the magnitude of its ripple (ΔV_{dd}) are determined by



(a)

(b)

Figure 3.2: Illustration of the DC-DC converter we employed.

$$
\begin{aligned}
V_{dd\_avg} &= DV_{ex} - R_L I_{load}, &(3.1)\\
\Delta V_{dd} &= \frac{DV_{ex}(1-D)}{8LCf_s^2}, &(3.2)\\
D &= (V_{dd} + R_L I_{load})/V_{ex}, &(3.3)\\
\varepsilon_b &= \frac{P_{load}}{P_{load} + P_b}, &(3.4)
\end{aligned}
$$

where V_{ex} is the external input voltage, I_{load} is the DC load current, D and f_s are respectively the duty cycle and the switching frequency of the pulse-width modulation signal, and R_L is the serial resistance of the inductor. By dynamically adjusting the duty cycle D through a set of sensing and controlling circuitry, V_{dd} can be sustained around the desired value. shifting from one V_{dd} to another in the DVS system is then achieved by programming the control circuitry to produce a new desired D, according to the DVFS controller's command. The ΔV_{dd} ,which is a source of on-chip power noise, is controlled by properly selecting the inductor L, the capacitor C and the switching frequency f_s.

The success of DVFS schemes is based on the assumption that the DC-DC converter maintains good power efficiency over the entire voltage scaling range with sufficiently fast transition between different output voltages. Faster transition time of the DC-DC converter is usually at the cost of higher power loss and the power loss varies with its output voltage. A different output voltage may result in a different DC voltage conversion power loss.

The dominant part of the power loss by the DC-DC converter illustrated in Fig. 3.2 is due to the power devices, namely the PMOS and NMOS switches, the switch driver, the inductor and the capacitor [58], and can be approximated by

$$\begin{aligned}
P_{loss} &\approx I_{load}^2 \left[DR_{PMOS} + (1-D)R_{NMOS} + R_L\right] \\
&+\frac{1}{3}\left(\frac{\Delta I_L}{2}\right)^2 \left[DR_{PMOS} + (1-D)R_{NMOS} + R_L + R_C\right] \\
&+\frac{1}{2}C_{SW}V_{ex}^2 f_s, \qquad (3.5) \\
\Delta I_L &= \frac{V_{ex}D(1-D)}{Lf_s}, \qquad (3.6)
\end{aligned}$$

where, as illustrated in Fig. 3.2(b), ΔI_L is the inductor current fluctuation, R_{PMOS}, R_{NMOS} and R_C are the serial resistance associated respectively with the PMOS switch, NMOS switch, and the output capacitor, and C_{SW} is the sum of the switch capacitance associated with the two switches and the switch driver.

As inferred in (3.3)–(3.6), the power loss is one value of which leads to the quadratic function of the duty cycle D which has a peak power loss; as the actual D deviates from that spot, the power loss decreases. Therefore, a different V_{dd} results in a different D and hence a different DC voltage conversion power loss. In conjecture with the preceding discussion on transition time, this fact implies the need for joint optimization of the DC-DC converter and the DVFS policy.

Furthermore, the power loss is highly correlated to the voltage transition time. The first-order model of the output voltage transition time (T_t) of the converter is expressed as [58, 59]

$$\begin{aligned}
T_t &= \frac{8Q_0}{\omega_0}, \qquad (3.7) \\
Q_0 &= \frac{1}{R}\sqrt{\frac{L}{C}}, \qquad (3.8)
\end{aligned}$$

where Q_0 and ω_0 are respectively the quality factor of the LC circuitry and the crossover frequency of the feedback control loop. For a smaller T_t, extending ω_0 and decreasing Q_0 are two effective ways. On the one hand, it is well known to converter designers that ω_0 is closely related to the switching frequency f_s. And the influence of increasing f_s on the power loss is double-sided: the part of power loss due to charging/discharging the switch capacitance C_{SW} is proportional to f_s, but increasing f_s also quadratically reduces inductor current ripple amplitude and hence results in less power loss caused by the parasitic resistance in the switches, inductor and capacitor. On the other hand, decreasing Q_0 implies increasing R_L and R_C as seen from (3.8), and hence increasing the power loss caused by those resistances.

The above complex and intertwined relationships between power loss, and output voltage and transition time clearly suggest that better balance between performance and energy saving can be achieved by pursuing cross-layer optimization.

3.1.3 System-Level Design

As shown in Fig. 1.5, the design of a DVFS controller includes the operative period, the evaluation model, and the DVFS algorithm. The DVFS controller evaluates each operating point based on the evaluation model at the end of each operative period. According to the evaluation result, an operating point is selected through the DVFS algorithm. The processor will operate at the selected operating point (supply voltage and operating frequency) in the next operative period. In this paper, we adopt the evaluation model and DVFS algorithm proposed in [24] for analysis.

The operative period determines the grain size of DVFS. Generally, finer-grained DVFS can better track workload variation and bring more benefits in energy saving. Hence, a DVFS controller can be improved through shortening the operative period. The minimum operative period is constrained by the output voltage transition time of the DC-DC converter. It means an operative period cannot be shorter than the voltage transition time, otherwise the transition cannot be finished in the operative period.

Table 3.1: The Working Set of Operating Points and Mapping from CPU usages to Expected Operating Points

OP	Voltage (V)	Frequency (GHz)	CPU Usage Interval	μ_m
1	0.9	0.5	0∼20%	0.1
2	1.0	0.625	20∼40%	0.3
3	1.1	0.750	40∼60%	0.5
4	1.2	0.875	60∼80%	0.7
5	1.3	1	80∼100%	0.9

An evaluation model referred as μ-map is utilized to measure energy consumption and the performance delay of each operating point in the working set. This model is proposed based on the fact that the optimal frequency/voltage that balances the energy consumption and the performance delay increases with the CPU usage μ. According the fact, the domain of CPU usage ($0 \leq \mu \leq 1$) is uniformly divided into N intervals. Each interval is represented by its mean value (center point) μ_m. The operating points from 1 to N (frequency/voltage from low to high) are sequentially corresponding to the usage intervals from low to high. In this case, a mapping from CPU usages to operating points is created. Tab. 3.1 shows the working set and the evaluation model used in this paper. An operating point is evaluated through the comparison between CPU usage and μ_m. If $\mu_m > \mu$, it indicates that the frequency of the operating point is higher than the optimal one and thereby the operating point suffers a penalty of energy consumption. Similarly, the operating point suffers a penalty of performance delay if $\mu_m < \mu$.

The online learning DVFS policy algorithm is given by Alg. 1 [24]. N indicates

ALGORITHM 1: Online Learning DVFS Algorithm

```
$N$ = number of operating points;
$\alpha \in [0,1]$, $\beta \in [0,1]$;
$w_i^{0} = \frac{1}{N}$ $(i = 1, 2, ..., N)$;
for operative periods $t = 1, 2, ...$ do
    1 $p_i^t = \frac{w_i^t}{\sum_{i=0}^{N} w_i^t}$;
    2 select operating point according to its probability factor in $p^t$;
    3 Operative period starts → apply selected operating point to on-chip
    circuit;
    4 Operative period ends →;
        $\mu$ = CPU usage;
        for operating point $i = 1, 2, ..., N$ do
            $l_{ei}^t = (\mu > \mu_{mi})$ ? $0 : (\mu_{mi} - \mu)$;
            $l_{pi}^t = (\mu > \mu_{mi})$ ? $(\mu - \mu_{mi}) : 0$;
            $l_i^t = \alpha l_{ei}^t + (1 - \alpha) l_{pi}^t$;
        end
    5 $w_i^{t+1} = w_i^t \beta^{l_i^t}$ ;
    6 t=t+1 ;
end
```

the number of operating points. α is a key learning policy parameter. β is a constant value used to update operating points' weights. μ indicates the average CPU usage of previous operative periods. μ_{mi} indicates μ_m corresponding to operating point (OP) i. w_i^t is the weight for OP i at the t-th DVFS operative period. p_i^t is the selection possibility of OP i. l_{ei}^t and l_{pi}^t respectively indicate the energy consumption penalty and the performance delay penalty of OP i during the t-th DVFS operative period. The DVFS controller can be optimized through turning the learning parameter α. As shown in Alg. 1, the weighted penalty of an operating point is given by

$$l_i^t = \alpha l_{ie}^t + (1 - \alpha) l_{ip}^t. \tag{3.9}$$

α is used to balance between energy consumption and performance delay in this equation. When α is close to 1, the energy consumption penalty l_{ei}^t is weighted more dominantly than the performance delay penalty l_{pi}^t. Hence, the operating points with higher frequencies have less opportunities to be employed. As a result, increasing α leads to a reduction of energy consumption and a growth of performance delay.

As discussed above, the system-level optimization can balance the CPU energy consumption and the performance delay. However, the energy consumption of the entire DVFS system may not be optimized through the system-level optimization only. The details are discussed in the following section.

3.1.4 Opportunities of Circuit/System Co-optimization

In this section, we identify key limitations of circuit-level only or system-level only optimization strategies and thereby identify cross-layer opportunities that can reduce the energy consumption and the performance delay. For most of DVFS designs, the DC-DC converter and the DVFS controller are designed separately. Even if the design trade-offs discussed above are properly balanced for each level, the entire

DVFS system may still not reach the overall optimality.

3.1.4.1 Limitations of System-Level Optimization

Without the design information of the DC-DC converter, a system-level optimization cannot optimize the total energy consumption.

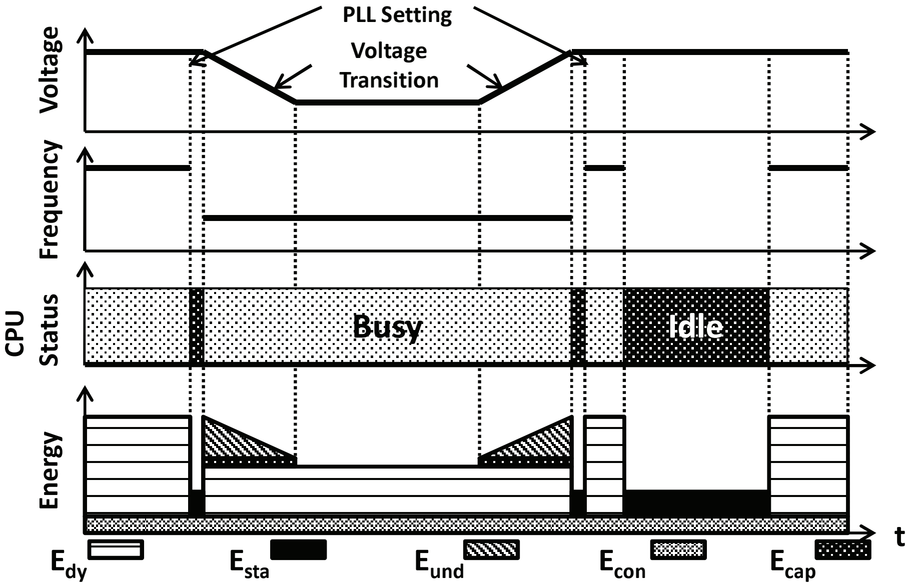


Figure 3.3: The supply voltage, frequency and energy consumption during a DVFS procedure. (E_{dy}, E_{sta}, E_{und}, E_{con}, and E_{cap} respectively represent the dynamic energy of the processor, the static energy of the processor, the under driving energy overhead during DVFS transition, the energy consumption of the DC-DC converter and the energy consumed by charging/discharging capacitors during voltage scaling.)

An exemplary DVFS procedure and related energy consumptions are shown in Fig. 3.3. This sequence is commonly used in modern microprocessors [20, 7]. The energy consumption appears during the typical DVFS procedure can be divided into two parts as follows.

$$E_{tot} = E_{proc} + E_{over}, \tag{3.10}$$

where E_{proc} is the energy consumed by the processor and E_{over} is the energy overhead of DVFS. E_{proc} is given as

$$E_{proc} = E_{dy} + E_{sta}, \tag{3.11}$$

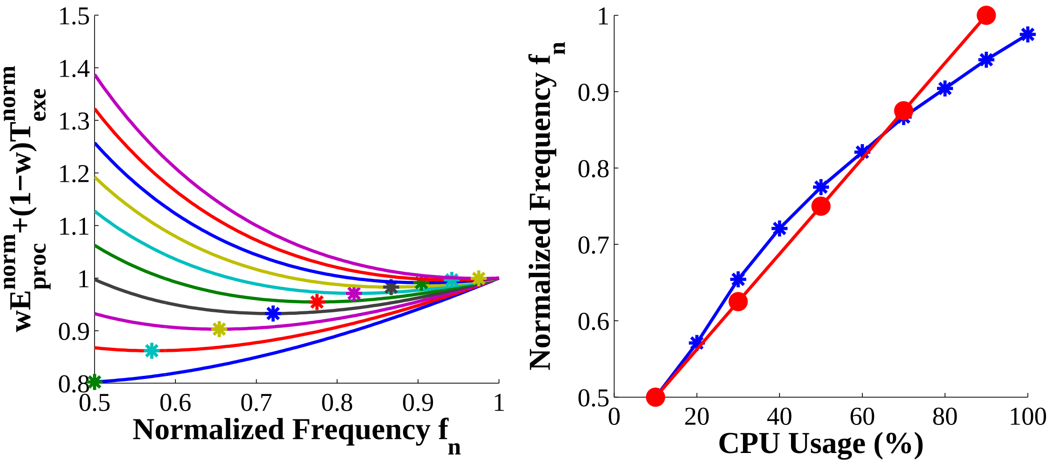
where E_{dy} and E_{sta} are respectively the dynamic energy and the static energy of the processor. The energy overhead of the DVFS is composed of three components as follows [60],

$$E_{over} = E_{con} + E_{und} + E_{cap}. \tag{3.12}$$

E_{con} is the energy loss of the DC-DC converter. It depends on the power efficiency of the DC-DC converter. It varies with the output voltage and the current loading of the DC-DC converter. E_{und} is the under driving energy. The clock frequency stays at the low level during both up-scaling and down-scaling voltage transitions. Hence, the processor works at a low frequency under an over-provisioned supply voltage and thereby E_{und} is consumed. E_{und} depends on the transition time between the two operating points T_t in (3.8) that is an important design parameter of the DC-DC converter. E_{cap} is the capacitance (processor decoupling capacitance and DC-DC converter equivalent output capacitance) charging/discharging energy during voltage transition.

The objective of the system-level optimization is to obtain a DVFS controller that balances the energy consumed by the processor E_{proc} and the performance delay T_{exe}. The objective can be achieved through improving the evaluation model, the DVFS policy (algorithm) and/or the operative period. However, the obtained controller may not be optimal for the total energy E_{tot}. The reasons are discussed as follows.

First, a system-level optimization is unaware of the DVFS energy overhead E_{over}. The performance delay and the energy consumption are evaluated based on the evaluation model that is a mapping from CPU statistic data (such as CPU usages,



(a) (b)

Figure 3.4: (a) The optimal frequencies to balance E_{proc} and T_{exe} with different μ. μ is the CPU usage when the workload is processed with the maximum frequency. The objective function is $wE_{proc}^{norm} + (1-w)T_{exe}^{norm}$. The technology node is 90nm. (b) μ-map from the CPU usage to the optimal frequency.

cache misses, and stall cycles) to expected operating points. The objective of the system-level optimization is to minimize both the processor energy E_{proc} and the execution time T_{exe}. Hence, the objective function is usually given as

$$f = wE_{proc}^{norm} + (1-w)T_{exe}^{norm}, \tag{3.13}$$

where $w \in (0,1)$ is the weight, $E_{proc}^{norm} = E_{proc}/E_{max}$, $T_{exe}^{norm} = T_{exe}/T_{max}$, E_{max} and T_{max} are respectively the processor energy and the execution time with the maximum frequency. Assume μ is the CPU usage when the workload is processed with the maximum frequency. With the given circuit design, there is an optimal frequency corresponding to each μ. As shown in Fig 3.4(a), the optimal frequency is marked with each CPU utilization. In this example, there are six operating points in the working set $f = 0.5, 0.6, ..., 1$. These optimal frequencies highly depend on the CPU utilization as shown in Fig. 3.4(b). Based on this relationship, the evaluation model (μ-map) in Alg. 1 is built. The CPU usages are be easily monitored and accessed through performance counters. E_{proc} is load dependent and thereby can be estimated through CPU statistic data. Unlike E_{proc}, E_{over} highly depends on circuit design parameters, such as the efficiency of the DC-DC converter, the decoupling capacitance and the voltage transition time. Hence, E_{over} cannot be estimated through performance counters only. Therefore, improving the evaluation model can better reflect E_{proc}, but it is not helpful for the evaluation of E_{over}.

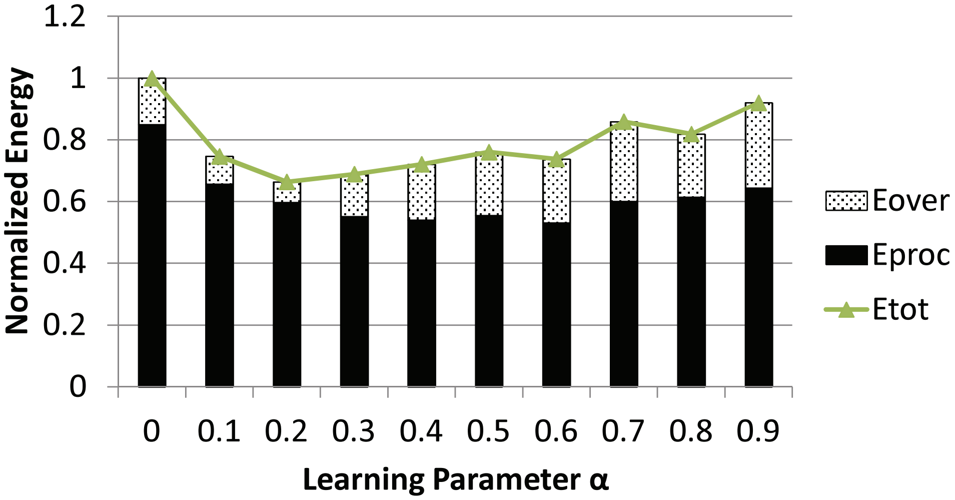


Figure 3.5: E_{proc} (the processor energy) and E_{over} (the DVFS energy overhead) may be tuned towards opposite direction through adjusting the DVFS policy. The DC-DC converter design and the DVFS operative period are fixed at different α values. The energy consumption is normalized to the total energy when the processor constantly operates at highest voltage/frequency ($\alpha = 0$). The simulation is based on benchmark *blackscholes* [2] with a single thread running on a single-core processor.

Second, adjusting the DVFS policy may tune E_{proc} and E_{over} in opposite directions. For example, when the optimizer adjusts α in the hope of reducing E_{proc} so as to reduce the overall system energy dissipation E_{tot}, the unawareness of E_{over} can produce misleading results. Fig. 3.5 shows the simulated energy dissipation and the execution time of benchmark *blackscholes* with a single thread processed on a single-core processor. As can be seen, increasing α from 0 to 0.6 reduces E_{proc} while E_{over} may increases. Finally, the total energy consumption E_{tot} may be increased unexpectedly.

In addition, shortening the operative period may lead to an unexpected growth of E_{over}. Fig. 3.6 shows the DVFS operative period's influence on the energy and the performance delay. Energy-delay production (EDP) is used to measure the trade-offs between the total energy and the performance delay. The energy-delay production (EDP) is given as,

$$EDP = E_{tot} \times T_{exe}, \tag{3.14}$$

where E_{tot} is the total energy consumption and T_{exe} is the run time. In Fig. 3.6, E_{tot} and T_{exe} are respectively normalized to the total energy and the run time when the processor constantly operates at highest voltage/frequency. Ideally, finer-grained DVFS can better track the workload variation and balance the energy and the performance delay. However, the EDP starts increasing when the operative period is shorter than 100K core cycles. This is because that shortening operative period leads to frequent voltage transitions and thereby increases E_{over} unexpectedly. Furthermore, the system-level optimization cannot be aware of this growth that may reduce or even overwhelm the benefit of the fine-grained DVFS.

In consequence, the DVFS controller obtained through a system-level optimization may not be suitable for the given DC-DC converter. The root cause of this problem is that the circuit-level design information is ignored in the system-level optimization.

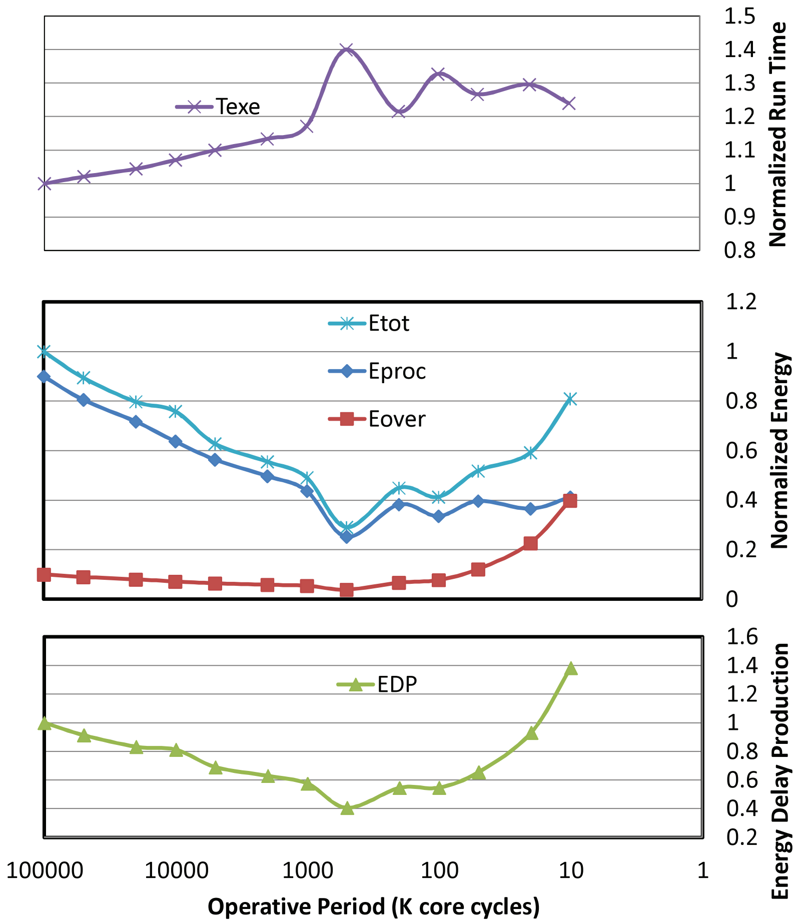


Figure 3.6: The influence of the DVFS operative period on the run time and the energy consumption. The learning parameter α is set to 0.5. The longest voltage transition time of the DC-DC converter is 9 μs (9K CPU cycles at the highest frequency). The energy consumption and the run time are respectively normalized to the total energy and the execution time when the processor constantly operates at the highest voltage/frequency. The simulation is based on benchmark *blackscholes* [2] with a single thread running on a single-core processor. The technology node is 90nm.

3.1.4.2 Limitations of circuit-level optimization

The objective of a circuit-level optimization is usually to design a DC-DC converter that minimizes the power loss, the area, and the voltage transition time. However, the result of an isolated circuit-level optimization may not be optimal when it working with the given DVFS controller.

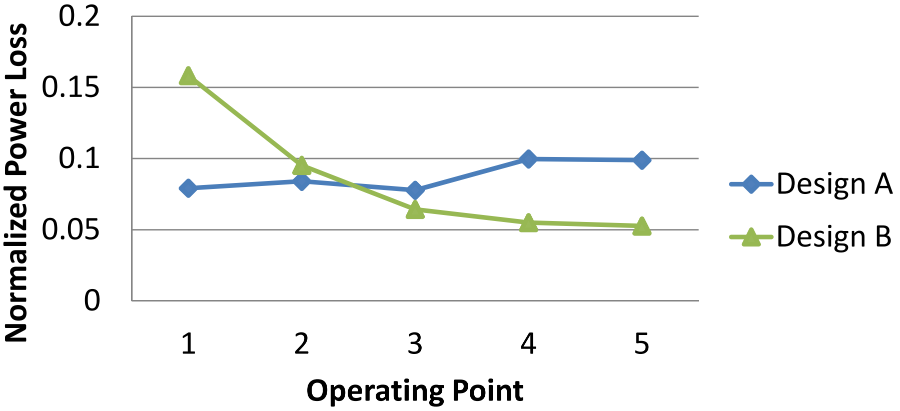


Figure 3.7: Power losses of two DC-DC converter designs at different operating points. The operating points are listed in Tab. 3.1. The power losses are normalized to the output power at OP 5. The simulation is based on benchmark *blackscholes* running with a single thread running on a single-core processor.

First, a circuit-level optimization may not optimize the total energy consumption. Minimizing the power loss is usually an objective of the circuit-level optimization. However, the power loss of a DC-DC converter is a function of output power that is determined by the operating point (voltage/freqeuncy). Fig. 3.7 presents the power losses of two DC-DC converter designs at the five operating points listed Tab. 3.1. As can be seen, design A's worst power loss and average power loss are both smaller than the ones of design B. In the perspective of energy efficiency, design A is closer to the result of a circuit-level optimization. However, design A may not be suitable for all DVFS controllers. For example, a DVFS controller that pursues low-α policy (Alg. 1) tends to select the operating points with higher voltages/frequencies. At these operating points (OP *3*, *4* and *5*), design A consumes more energy than design B. Hence, design A may not be optimal for a low-α DVFS policy even if it gets more credits in circuit-level optimization.

Second, a circuit-level optimization only can easily lead to overdesign without considering the DVFS controller. A circuit-level optimization provides a trade-off among different performance parameters. But the trade-off cannot be optimal when the DC-DC converter works for certain DVFS controllers. For example, a circuit-level optimization may increase power loss and/or area to shorten the output voltage transition time for finer-grained DVFS. However, the sacrifice of power and area could be meaningless for certain DVFS controllers. For an instance, if the operative period is not fine enough or the DVFS policy prefers operating points in high power loss zone, the DC-DC converter design may bring negative benefit and it may turn out to be over designed for the controller.

In conclusion, the DVFS controller and the DC-DC converter respectively obtained from isolated circuit-level and system-level optimizations may not be suitable for each other. Therefore, the targeted cross-layer optimization is necessary in order to obtain the optimal DVFS system.

3.1.5 General Hierarchical Co-optimization

The limitations of the circuit-only and policy-only optimization as discussed in the pervious section have motivated us to develop a co-optimization strategy. While this cross-layer approach is inherently computationally demanding, we develop a hierarchical two-step methodology to alleviate the complexity of the co-optimization as shown in Fig. 3.8.

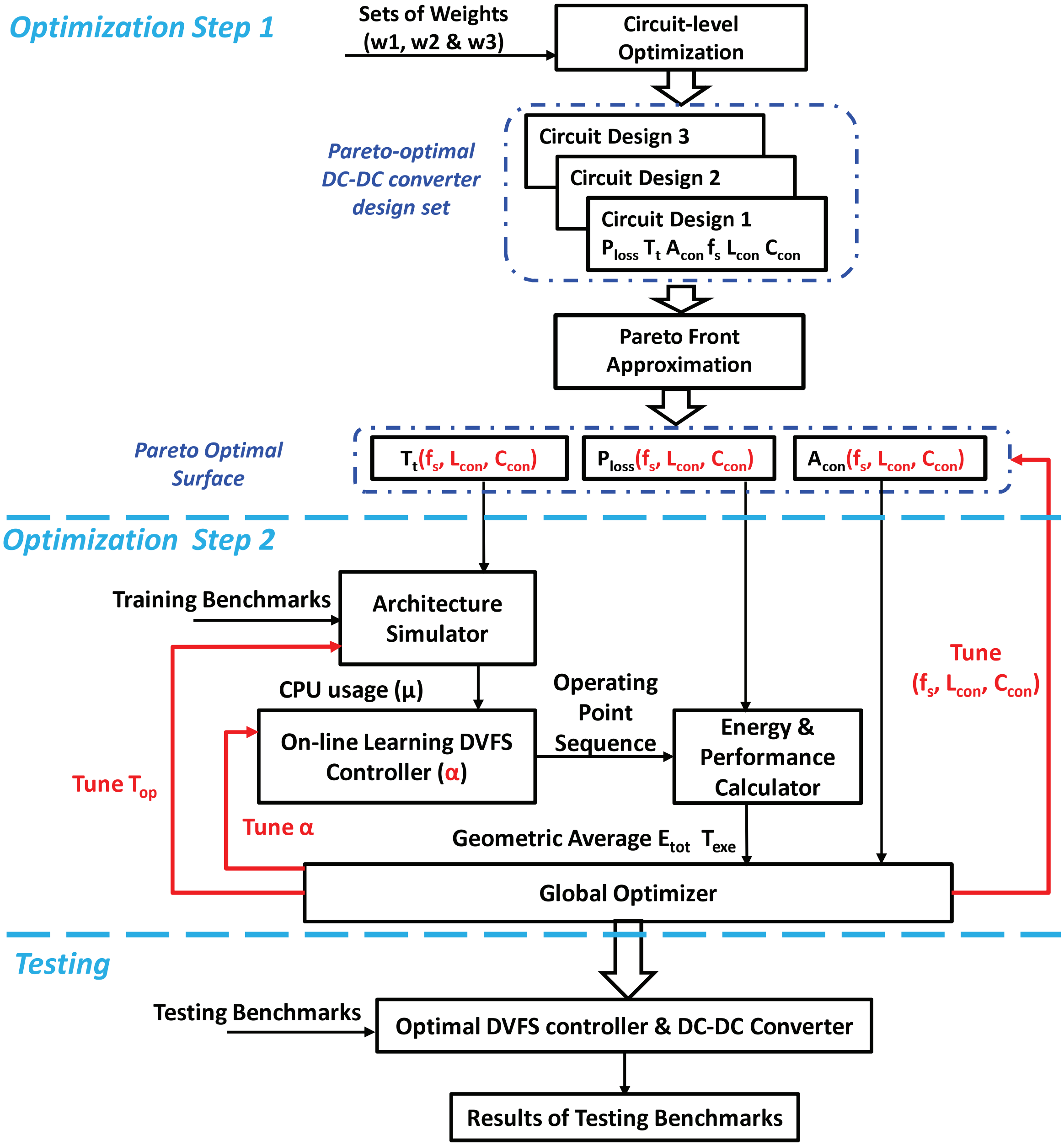

Optimization Step 1
Sets of Weights (w1, w2 & w3)
Circuit-level Optimization
Pareto-optimal DC-DC converter design set
Circuit Design 3
Circuit Design 2
Circuit Design 1
P_{loss} T_t A_{con} f_s L_{con} C_{con}
Pareto Front Approximation
Pareto Optimal Surface
$T_t(f_s, L_{con}, C_{con})$
$P_{loss}(f_s, L_{con}, C_{con})$
$A_{con}(f_s, L_{con}, C_{con})$
Optimization Step 2
Training Benchmarks
Architecture Simulator
CPU usage (μ)
On-line Learning DVFS Controller (α)
Operating Point Sequence
Energy & Performance Calculator
Tune (f_s, L_{con}, C_{con})
Tune T_{op}
Tune α
Geometric Average E_{tot} T_{exe}
Global Optimizer
Testing
Testing Benchmarks
Optimal DVFS controller & DC-DC Converter
Results of Testing Benchmarks

Figure 3.8: Hierarchical circuit-level and system-level co-optimization and testing flow.

The first step in this flow is geared to perform circuit-level optimization for the DC-DC converter. Instead of producing a single optimal DC-DC converter design, we find a set of pareto-optimal converter designs. This allows us to find a set of promising converters without invoking expensive full-system analyses. These pareto-optimal designs are fed to the second step to complete the cross-layer optimization of the full system.

As the first step of our optimization strategy, the circuit-level optimization finds an optimal set of trade-offs for the DC-DC converter in terms of power loss, transition time, and area overhead. The latter one is reflected by the realistic values of the inductor and capacitor representing the areas of these components on the PCB. Note that it is important to consider the noise effects created by the output ripples of the DC-DC converter as a design constraint. To be more specific, one needs to consider not only the amount of voltage ripples created immediately at the output of the DC-DC converter, but also the propagation of the ripples through the package and on-chip power distribution and finally the ripple-induced noises seen by the on-chip devices.

The package model of the processor is modeled as RCL network [11]. The on-chip power delivery network is modeled as a resistive network that is composed of two power domains. Each domain is a resistive grid with 100,000 nodes. With this model of passive power distribution, frequency-domain AC analysis is performed to obtain the transfer functions from the output of the DC-DC converter to various output nodes on the on-chip power distribution network over a typical range of converter switching frequency. Finally, the amplitude of the ripples on the on-chip device side are characterized by multiplying the corresponding transfer functions with the amplitude of the converter output ripple at the switching frequency.

The optimization of the DC-DC converter design is performed by tuning the design parameters including switching frequency (f_s), the lossy inductor (L_{con}), and the lossy capacitor (C_{con}) to arrive at the minimum of the objective function, i.e., a weighted sum of the aforementioned three figure-of-merits on the premise of not exceeding the upper limit of the amount of V_{dd} ripple. The objective function is

$$\begin{aligned} min. \quad & f_{DC-DC} = w_1 P_{loss\_WC} + w_2 T_{t\_WC} + w_3 A_{con}, \\ s.t. \quad & |H_{PG}(f_s)|\Delta V_{dd} \le UB_{pnoise}, \end{aligned} \tag{3.15}$$

where w_i ($i = 1, 2, 3$) are the weights, P_{loss_WC} represents the power loss under the worst-case output voltage, T_{t_WC} represents the longest transition time between two DVFS voltages, A_{con} is the sum of the area occupied by the inductor and the capacitor in the DC-DC converter, $H_{PG}(f_s)$ is the worst-case transfer function from the output of converter to the power grid nodes evaluated at the switching frequency for calculating the amount of on-chip power noise induced by ΔV_{dd}, and UB_{pnoise} is the specified upper bound for that noise.

By setting different values of weights w_i, we obtain several optimized DC-DC converter design points on the Pareto surface. A Pareto-optimal design point has at least one specification that is better than the corresponding one of any other designs. We obtain an approximated continuous Pareto surface though interpolation. The surface provides a design set in which the global optimizer searches for the optimal circuit design through turning f_s, L_{con}, and C_{con}. Each design point on the surface corresponds to a set of performance parameters including the longest voltage transition time T_t, and the DC-DC converter area overhead A_{con}. The Pareto surface is modeled as a set of circuit-level design constraints for the global optimization in Step 2.

In Step 2, the architecture simulator determines the DVFS operative period that is constrained by the longest voltage transition time T_t. The architecture simulator simulates the training benchmarks and generates a series of CPU usages corresponding to the series of DVFS operative periods. The online learning DVFS controller given in Alg. 1 with $\alpha \in (0,1)$ is employed to generate a series of operating points according to the CPU usages. Based on the operating points and the power loss P_{loss} of the DC-DC converter, the energy/performance calculator estimates the geometric average total energy E_{tot} and geometric average execution time T_{exe} of training benchmarks. The global optimizer updates the results and tunes α, f_s, C_{con}, and L_{con} to find the optimal DVFS policy and the optimal DC-DC converter design.

Subject to the (optimal) circuit-level design constraints based on the Pareto surface, the objective function of the global optimization is formulated given by

$$\begin{aligned} min. \quad & f = w_e E_{tot} + w_p T_{exe} + w_a A_{con}, \\ s.t. \quad & \begin{cases} 1 \geq \alpha \geq 0, \\ T_{op} \geq T_{t\_WC}, \end{cases} \end{aligned} \tag{3.16}$$

where w_e, w_p, and w_a are respectively the weights of geometric average total energy, geometric average performance delay, and area overhead, E_{tot} represents the geometric average total energy consumption of training benchmarks, T_{exe} represents the geometric average execution time of training benchmarks, A_{con} is the area overhead of the DC-DC converter, α is the key learning parameter of DVFS algorithm, T_{op} represents the operative period of DVFS, and T_{t_WC} indicates the longest transition time of the DC-DC converter. The optimization flow is not restricted to any specific objective function but can use any generic function of E_{tot}, T_{exe}, and A_{con}. The formulation of the objective function can be selected by designers.

The co-optimization finally generates one optimal DVFS controller and one optimal DC-DC converter. In the testing step, the benchmarks in the testing set are used to evaluate the obtained designs. The experimental results of testing benchmarks are shown in the following section.

3.1.6 Experimental Results

The system-level only optimization, circuit-level only optimization and system/circuit co-optimization are compared in this section. The results are based on the simulations on single-core processor.

3.1.6.1 Experimental Setting

The setups of the experiments are tabulated in Tab. 3.2. The structure of the DC-DC converter is shown in Fig. 3.2(a). The DVFS policy is given by Alg. 1. The DVFS online learning controller and energy/performance calculator are implemented by C++. The circuit-level and global optimizers are implemented by a nonlinear optimization problem solver APPSPACK (Asynchronous Parallel Pattern Search Package) [52]. The benchmarks of PARSEC 2.1 [2] are used as workloads to train and test different strategies. The training set is composed of *blackscholes, bodytrack, canneal, facesim, freqmine, raytrace, vips.* The testing set is composed of *dedup, ferret, fluidanimate, raytrace, streamcluster, swaptions, x264.* The application domain of each set includes Financial Analysis, Animation, Data Mining, Media Processing, etc.

Four designs are referred in this section. They are described as follows.

Ref Reference design. For the DVFS controller, $\alpha = 0.5$, the operative period is 100K core cycles. For the DC-DC converter, $f_s = 2MHz$, $L_{con} = 100nH$, and $C_{con} = 5\mu F$. This design is used as a reference for comparison. The energy, the execution time, and the area overhead of other designs are all normalized to the values of this design.

Table 3.2: Experimental Setting of DVFS design for Single-Core Processor

Circuit Level		System Level	
Size of PDN	200K	Number of Cores	1
On-chip Decaps	200nf	ISA	ALPHA
Simulator	Cadence Spectre	L1 I-Cache	32KB
Optimizer	APPSPACK [52]	L1 D-Cache	32KB
Technology	90/45/22nm[1]	L2 Cache	128KB
		Execution Model	Out of order
		Working Set	see Tab. 3.1
		Simulator	gem5 [61]
		Benchmark[2]	PARSEC 2.1 [2]

[1] The DC-DC converter is designed at 90nm technology node; the processor is designed at 90/45/22nm technology nodes respectively.

[2] Training Set:*blackscholes, bodytrack, canneal, facesim, freqmine, vips.* Testing Set: *dedup, ferret, fluidanimate, raytrace, streamcluster, swaptions, x264*

S-only Design obtained by the system-only optimization. The DVFS controller is optimized through the isolated system-level optimization. The DC-DC converter design is the same as the reference design.

C-only Design obtained by circuit only optimization. The DC-DC converter is optimized through the isolated circuit-level optimization. The DVFS controller is the same as the reference design.

Co-op Design obtained by our proposed strategy. The DVFS controller and the DC-DC converter are co-optimized through our proposed cross-layer co-optimization.

The optimization and testing flow of S-only strategy is shown in Fig. 3.9. System-only optimization take the DVFS controller as the design target. The architecture simulator simulates the training benchmarks and generates a series of CPU usages corresponding to the series of DVFS operative periods. The online learning DVFS controller given in Alg. 1 with $\alpha \in (0, 1)$ is employed to generate a series of operating

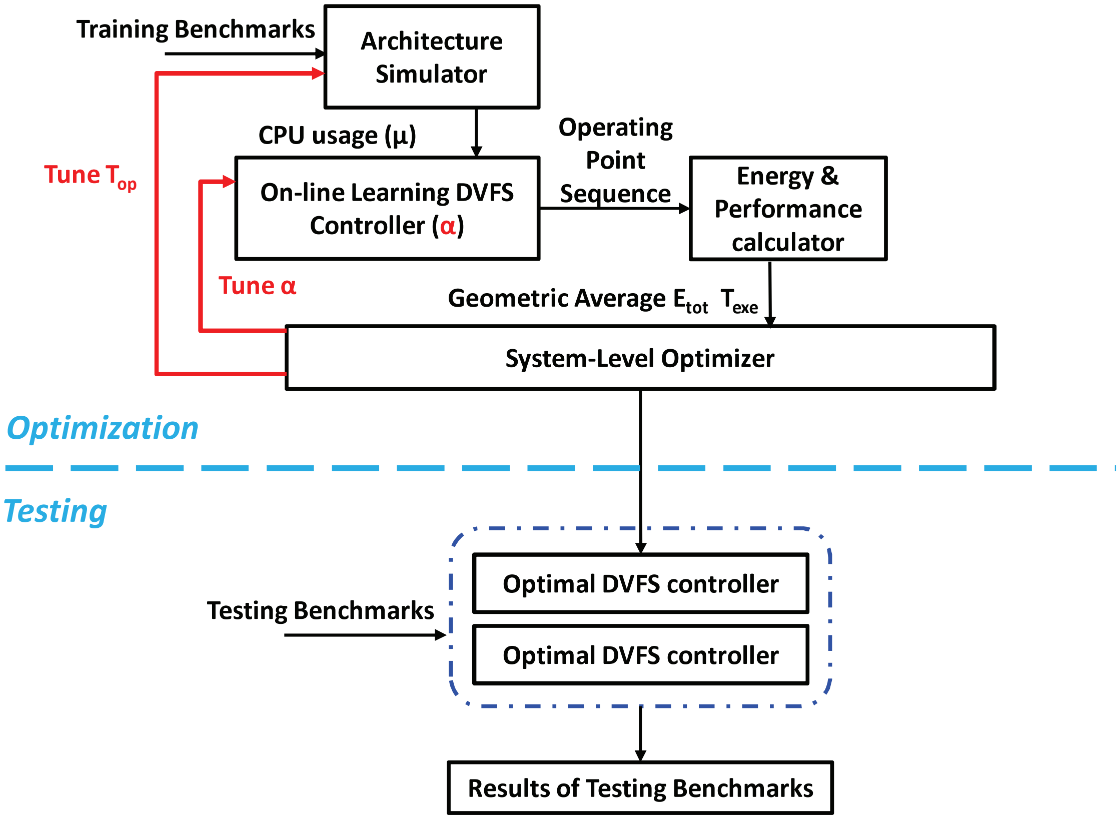


Figure 3.9: System-only optimization and testing flow.

points according to the CPU usages. Based on the operating points and the power loss P_{loss} of the DC-DC converter, the energy/performance calculator estimates the geometric average total energy E_{tot} and geometric average execution time T_{exe} of training benchmarks. The system-level optimizer tunes learning parameter α and DVFS operative period T_{op} to find the optimal DVFS controller design. The objective function and the constraints of system-level optimizer are given as follows.

$$\begin{aligned} min. \quad & f = w_e E_{tot} + w_p T_{exe}, \\ s.t. \quad & \begin{cases} 1 \geq \alpha \geq 0, \\ T_{op} \geq T_{op_max}, \end{cases} \end{aligned} \tag{3.17}$$

where w_e and $w_p$are respectively the weights of geometric average total energy and performance delay of training benchmarks, E_{tot} represents the geometric average total energy consumption, T_{exe} represents the geometric average execution time, α is the key learning parameter of DVFS algorithm, T_{op} represents the operative period of DVFS, and T_{op_max} indicates lower bound of T_{op}. The optimization flow is not restricted to any specific objective function but can use any generic function of E_{tot} and T_{exe}. The formulation of the objective function can be selected by designers. The optimization finally generates one optimal DVFS controller. The obtained DVFS controller and the DC-DC converter in the reference design are used to test the benchmarks in the testing set.

The optimization and testing flow of C-only strategy is shown in Fig. 3.10. Circuit-only optimization take the DC-DC converter as the design target. The circuit simulator simulates the given DC-DC converter design to generate evaluation parameters that include output longest voltage transition time $T_{t_{WC}}$, largest power loss $P_{loss_{WC}}$, area overhead A_{con}. The circuit-level optimizer tunes the switching fre-

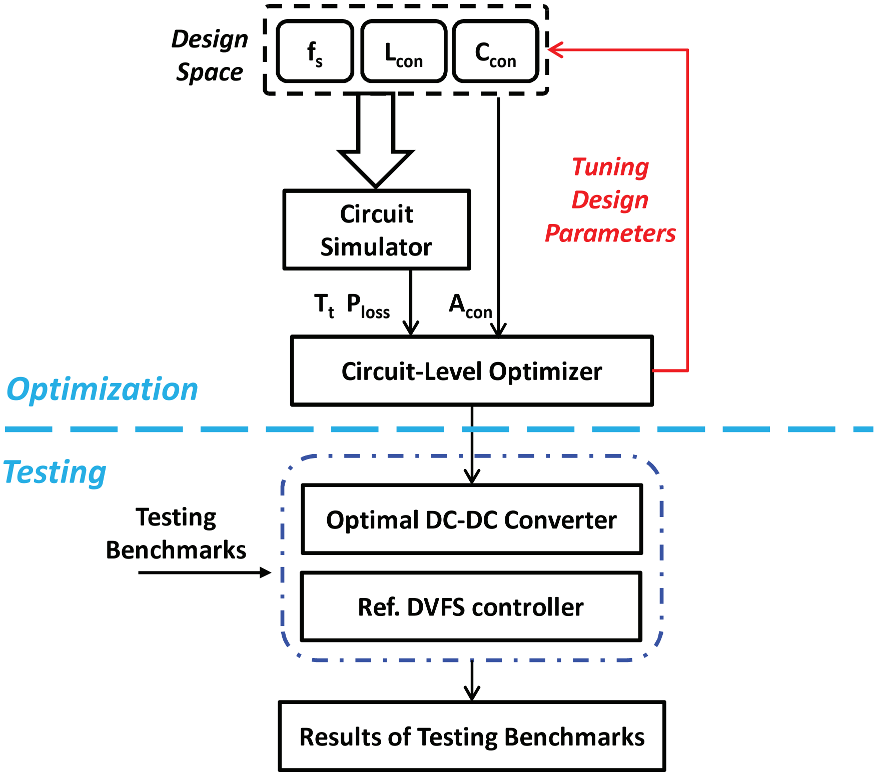


Figure 3.10: Circuit-only optimization and testing flow.

quency f_s, the lossy inductor L_{con}, and the lossy capacitor C_{con} to find the optimal DC-DC converter design. The objective function and the constraints of circuit-level optimizer are given by (3.15). The optimization flow is not restricted to any specific objective function but can use any generic function of $T_{t_{W}C}$, $P_{loss_{W}C}$, and A_{con}. The formulation of the objective function can be selected by designers. The optimization finally generates one optimal DC-DC converter. The obtained DC-DC converter and the DVFS controller in the reference design are used to test the benchmarks in the testing set.

In the following section, we discuss and compare S-only, C-only, and CO-op strategies when they are applied to the processor at different technology nodes. The technology scaling significantly influences the normalized power consumption of the processor while it may not influence the normalized execution time. The reasons are as follows.

When a processor processes a task at the highest frequency f_{max}, the execution time is given by [62]

$$T_{max} = \mu + (1 - \mu), \tag{3.18}$$

where μ is the time during which the processor executes instructions and $1 - \mu$ is the stall time of the processor due to cache misses etc. Then, the normalized execution time when the processor operates at frequency f is given by

$$\begin{aligned} T_n &= \frac{T}{T_{max}} \\ &= \frac{\mu f_{max}}{f} + (1 - \mu) \\ &= \frac{\mu}{f_n} + (1 - \mu), \end{aligned} \tag{3.19}$$

where $f_n = \dfrac{f}{f_{max}}$ is the scaling factor of frequency (normalized frequency). μ is determined by the characteristic of the workload and f_n is determined by the specific work set of operating points. In our experiments, we assume that the processors at different technology nodes process the same training/testing benchmarks and use the same working set of operating points. Hence, the normalized execution time may not change as technology scales.

The average power consumption is a characteristic of the processor that highly depends on the technology. When a processor operates at the highest voltage/frequency operating point (V_{max}, f_{max}), the power can be estimated as follows [62].

$$\begin{aligned} P_{max} &= D_{max} + S_{max} \\ &= C_L V_{max}^2 f_{max} + I_L V_{max} \\ &= (1-\rho) + \rho, \end{aligned} \tag{3.20}$$

where D_{max} is the dynamic power, S_{max} is the static power, C_L is the equivalent loading capacitance of the processor. I_L is the average total leakage current of the processor, which is assumed to be constant within a typically small voltage scaling range. ρ is the ratio of static power to total power. Then, the normalized processor power consumption when the processor operates at at operating point (V, f) is given as

$$\begin{aligned} P_n &= \frac{D_{max}}{P_{max}} \times \frac{V^2 f}{V_{max}^2 f_{max}} + \frac{S_{max}}{P_{max}} \times \frac{V}{V_{max}} \\ &= (1-\rho) V_n^2 f_n + \rho V_n, \end{aligned} \tag{3.21}$$

where $V_n = \dfrac{V}{V_{max}}$ is the scaling factor of voltage (normalized voltage). f_n and V_n depend on the specific working set of operating points. ρ highly depends on the technology node. As technology scales, static power gradually become becomes a very significant part of total power. In this dissertation, we set that the ratios of static power to total power (ρ) at technology nodes 90nm, 65nm, and 22nm to be respectively 20%, 40%, and 50% [63, 64, 5].

3.1.6.2 Results of Circuit-Level Optimization

Tab. 3.3 shows the pareto-optimal design set obtained through circuit-level optimization in Step 1. The design set includes eight optimal DC-DC converter designs that are used for pareto-optimal surface interpolation.

Table 3.3: Obtained Pareto-Optimal DC-DC Converter Design Set

No.	P_{loss_WC}(mW)	T_{t_WC}(μs)	A_{con}	f_s (MHz)	L_{con}(nH)	C_{con}(μF)
1	165.9	2.6	2.24e-6	12.6	156	2.1
2	422.8	264	1.10e-7	18	65	0.05
3	44.5	48.2	5.10e-6	2	100	5
4	140.8	9	1.19e-6	10	97	1.1
5	186.5	2.57	3.75e-6	14.8	100	3.65
6	51.2	22.9	2.96e-5	1.2	200	29.4
7	90.9	5.8	2.95e-5	5.8	52	29.3
8	69.0	51.6	2.08e-6	1.9	322	1.75

3.1.6.3 Results of Co-Optimization at 90nm Technology Node

We first present the results at 90nm technology node. The system-only optimization (S-only), the circuit-only optimization (C-only) and our proposed cross-layer optimization (Co-op) are respectively applied to a single-core processor.

Fig. 3.11 shows geometric average processor energy E_{proc} and geometric average DVFS energy overhead E_{over} based on the testing set with different strategies. The energy consumptions are normalized to the geometric average energy of the reference

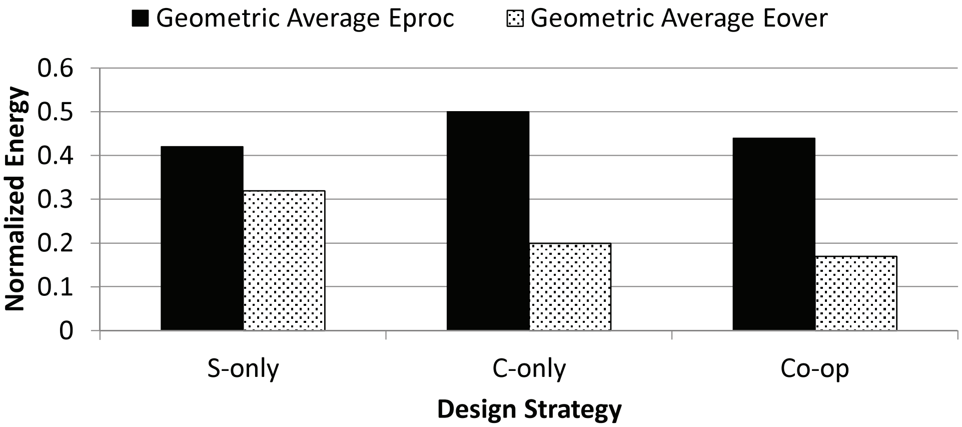
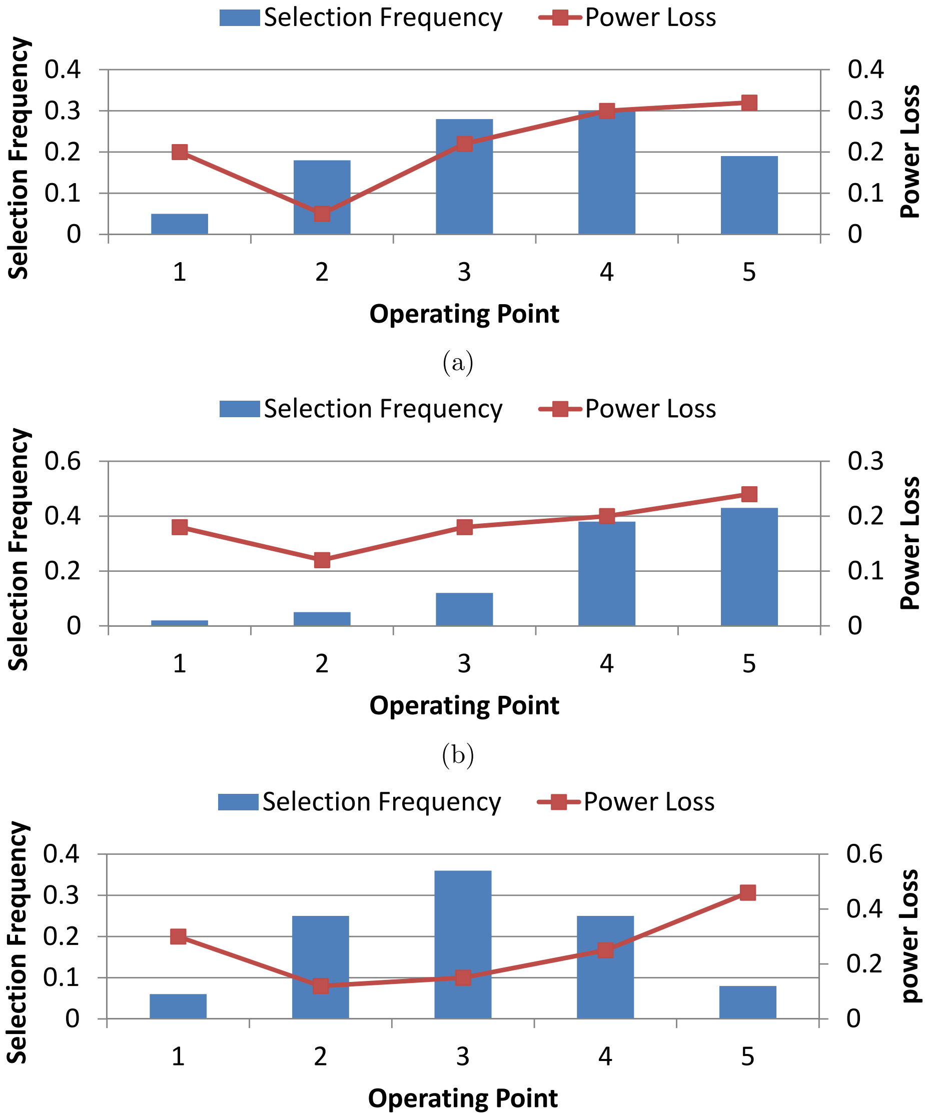


Figure 3.11: Normalized geometric average energy consumptions of the testing set. For each strategy, the geometric average processor energy and the geometric average DVFS energy overhead are normalized to the geometric average total energy of the reference design. The circuit-level and system-level designs are obtained based on the training set with S-Only, C-Only, and Co-Op strategies respectively. For each benchmark, the simulation is carried out with a single thread processed on a single-core processor.

design. Compared with the reference design, system-level only and circuit-level only optimization both reduce the total energy consumption. For the system-level only optimization, the reduction is mainly due to the decrease of E_{proc}. For the circuit-level only optimization, the reduction mainly comes from the decrease of E_{over}. Compared with those two strategies, co-optimization reduces both E_{proc} and E_{over}.

The reason for the results is based on the DC-DC converter's power loss and the selection frequency of different operating points as shown in Fig. 3.12. The system-level only (S-Only) optimization improves the DVFS controller to better track the workload. Hence, E_{proc} is reduced through selecting more suitable operating points. However, the system-level optimization does not consider the design of the DC-DC converter. As can be seen in Fig. 3.12(a), the power loss of the DC-DC converter is high at operating points (e.g. OP $3 \sim 5$) that are selected frequently by the DVFS policy. As a result, E_{over} is not obviously reduced. Compared with S-Only



(a)

(b)

(c)

Figure 3.12: Selection frequency and power loss at different operating points. The results are based on benchmarks of the testing set with single thread processed on a single-core processor. The circuit-level and system-level designs are obtained based on the training set with S-Only, C-Only, and Co-Op strategies respectively. (a) System Only Optimization; (b) Circuit Only Optimization; (c) Cross-layer Co-optimization.

optimization, circuit-level only optimization (C-Only) improves the average power loss of the DC-DC converter. Hence, E_{over} is obviously reduced. However, the design of the DVFS controller is kept the same as the reference design. This is the reason why E_{proc} is not improved. Our proposed co-optimization (Co-op) perfectly considers the design trade-offs at the system level and circuit level. As shown in Fig. 3.12(c), the DVFS policy is more suitable for the DC-DC converter design than the other two strategies. The controller can track the workload as well as the system-level only optimization. In addition, the power losses at the frequently selected operating points are reduced through the circuit optimization. Therefore, cross-layer co-optimization has an advantage over energy saving.

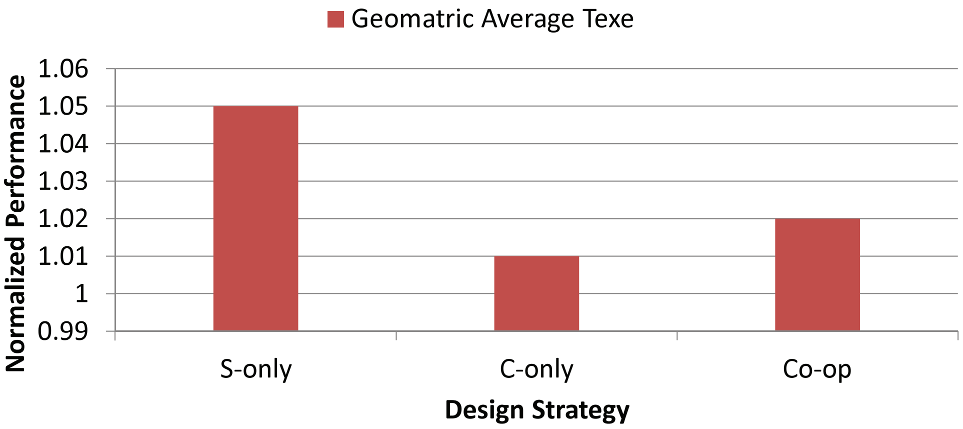
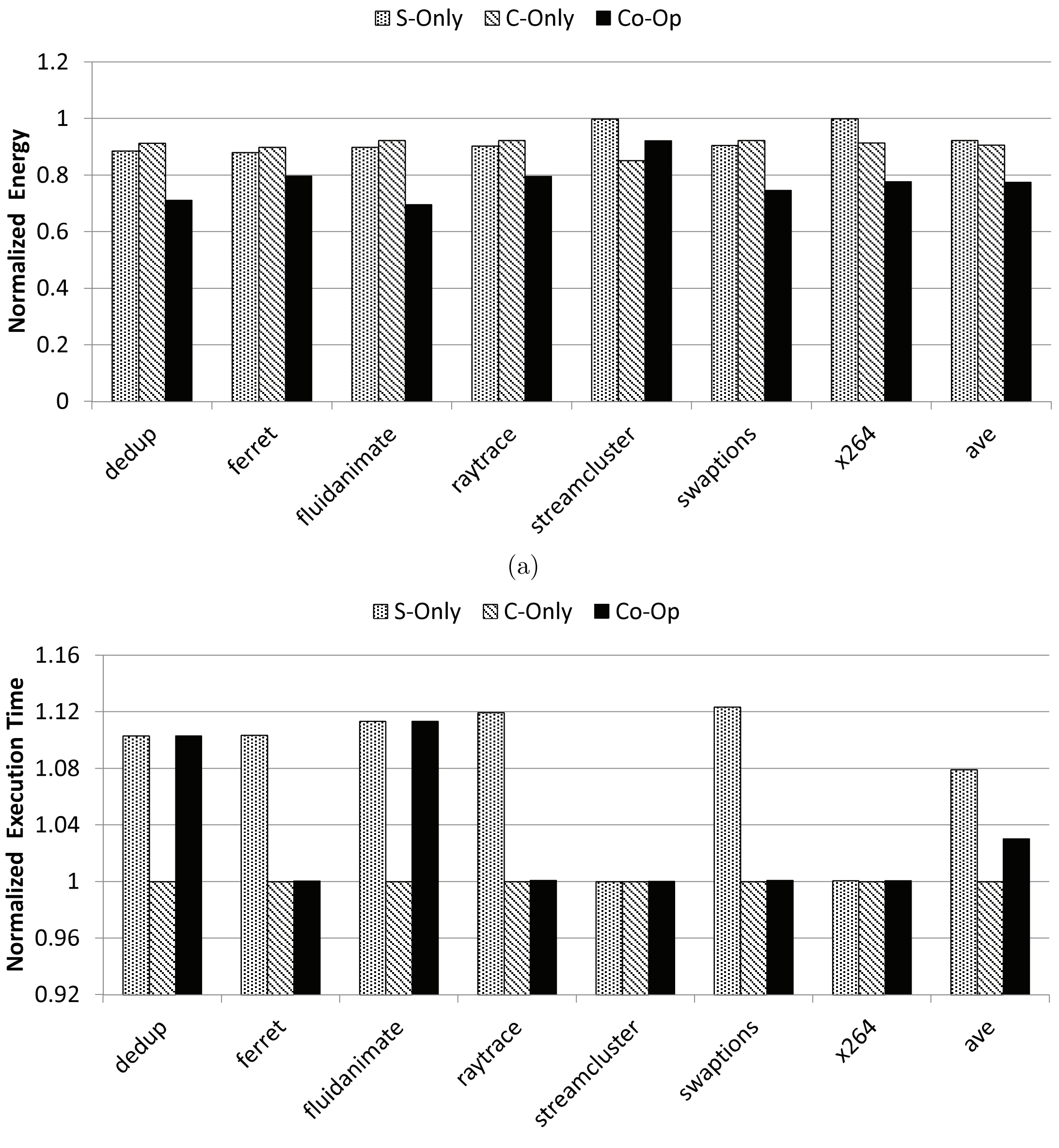


Figure 3.13: Normalized geometric average execution time of benchmarks of the testing benchmarks. For each strategy, the geometric average execution time is normalized to the geometric average execution time of the reference design. The circuit-level and system-level designs are obtained based on the training set with S-Only, C-Only, and Co-Op strategies respectively. For each benchmark, the simulation is carried out with a single thread processed on a single-core processor.

Fig. 3.13 shows the normalized execution time of benchmarks in testing set with different design strategies. The performance delay of system-level only optimization is relatively high. This is because the energy benefits of the strategy is based on the trade-off with performance delay through the DVFS policy. The circuit-level only and our proposed co-optimization has smaller performance delay since parts of the energy is saved through the improvement of the DC-DC converter design. The performance delay of co-optimization is larger than the circuit-level only optimization. This is because the co-optimization reduces the processor energy E_{proc} through low-frequency operating points that lead to a growth of performance delay.

Fig. 3.14(a) and 3.14(b) respectively show the energy consumption and execution time of PARSEC benchmarks with different strategies at 90nm technology node. S-Only Strategy utilized learning policy to balance energy consumption and performance delay. Hence, it generates 8% performance delay compared with reference design. The energy consumption is reduced by 8% compared with the reference design. The DC-DC converter cannot be optimized though S-Only Strategy. Hence, the energy saving is limited by the power loss of the DC-DC converter. C-Only strategy uses the same DVFS controller as the reference design. Hence, the performance of C-Only design is similar to the reference design. The geometric average energy consumption of C-Only design is reduced by 10% compared with the reference design. Co-Op Strategy optimizes the circuit-level and system-level designs together. Compared with system-level only optimization, the geometric average energy is reduced by 16%. Compared with circuit-level only optimization, the geometric average energy is reduced by 15%. Co-Op Strategy tunes the learning parameter to reduce the energy of processor. Hence, the geometric average performance delay is increased by 3% compared with C-Only design. But compared with S-Only design, the geometric average performance delay is reduced by 4.5%.



(a)

(b)

Figure 3.14: Normalized total energy consumption and execution time at 90nm technology node. The results are based on the benchmarks in the testing set. The circuit-level and system-level designs are obtained based on the training set with S-Only, C-Only, and Co-Op strategies respectively. The energy consumption and execution time of the three designs are normalized to the reference design. The last column shows the geometric energy/performance of the benchmarks. (a) Normalized total energy consumption. (b) Normalized execution time.

3.1.6.4 Results of Co-Optimization at Advanced Technology Nodes

In this section, we present the results at 45nm and 22nm respectively. As technology scales down, static power becomes dominant in total power consumption. However, DVFS cannot tune static power as efficiently as dynamic power. Hence, the interaction between performance delay and power consumption becomes tighter as technology scales down.

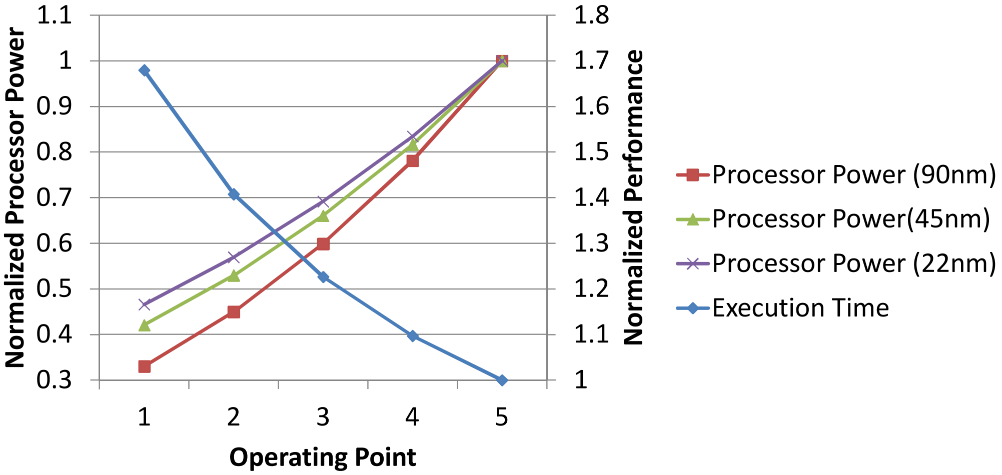


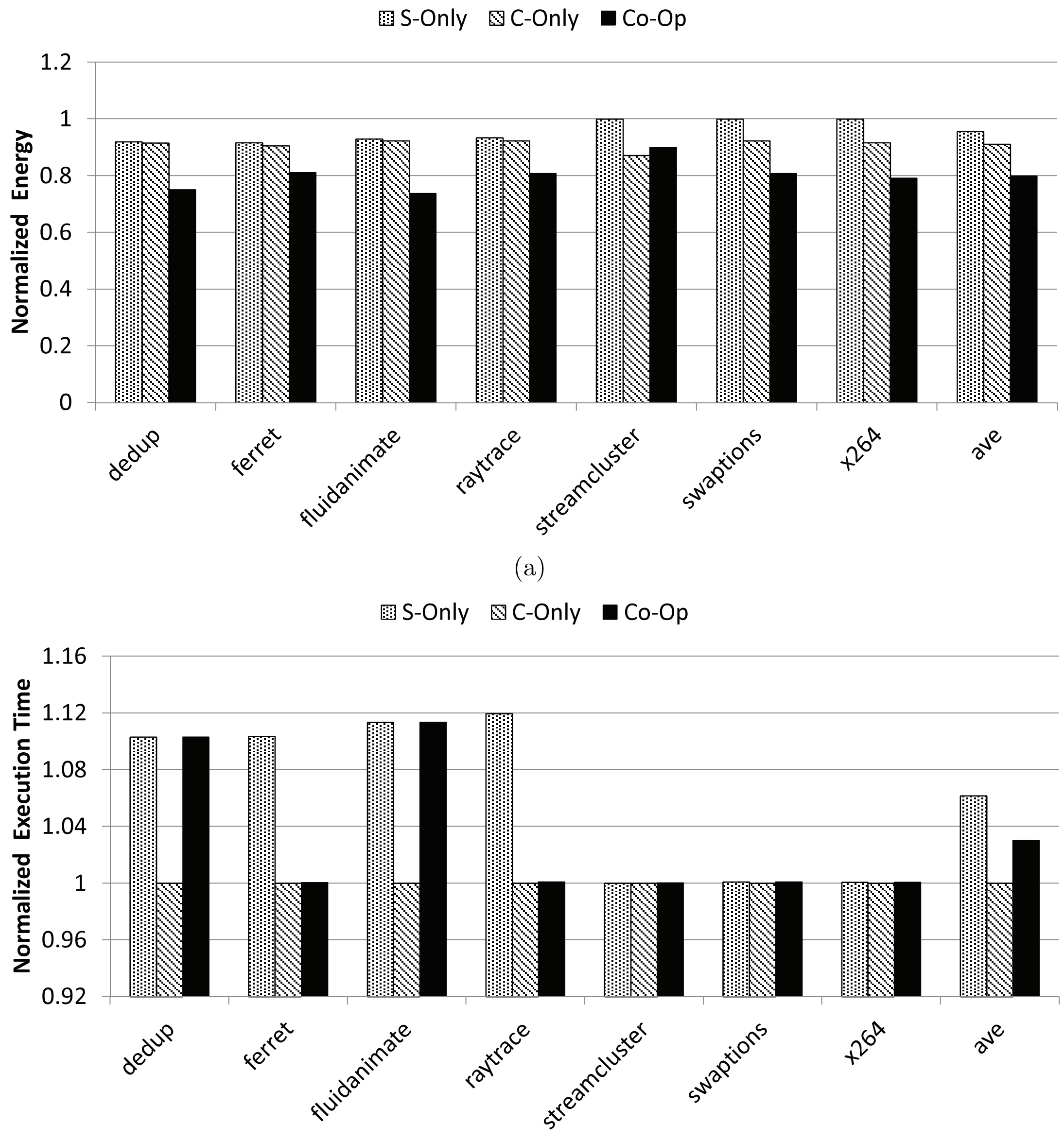
Figure 3.15: Execution time and processor energy consumptions at different operating points. The execution time at each operating point is normalized to the execution time at OP 5 (the highest voltage/freqeuncy). At each technology node, the processor energy consumption at each operating point is normalized to the processor energy at OP 5 (the highest voltage/freqeuncy). The results are based on benchmark *bodytrack* with one thread on a single-core processor.

Fig. 3.15 shows the execution time and processor power consumptions at different operating points. The same benchmark *bodytrack* is processed on processors at different technology nodes (90nm, 45nm, and 22nm). The execution time at each operating point is normalized to the execution time at OP 5 (the highest voltage/freqeuncy).

The processor power consumption at each operating point is normalized to the processor energy at OP 5 (the highest voltage/freqeuncy). The normalized execution time only depends on the normalized frequency (f/f_{MAX}) of each operating point. Hence, the normalized execution time does not vary with technology node. In contrast, the normalized processor power highly depends on the processing technology, for the ratio of static power to dynamic power varies with technology node. As technology scales down, the power consumption at same operating point increases. In other words, less energy can be saved with the same performance delay.

Fig. 3.16(a) and 3.16(b) respectively show the energy consumption and execution time of PARSEC benchmarks with different strategies at 45nm technology node. S-only Strategy generates 6% geometric average performance delay compared with the reference design. The energy consumption is reduced by 4% compared with the reference design. The requirement of performance limit its energy saving. The performance of C-only design is similar to the reference design since it uses the same DVFS controller as the reference design. The geometric average energy consumption of C-only design is reduced by 9% compared with the reference design. Co-op Strategy optimizes the circuit-level and system-level designs together. Compared with system-level only optimization, the geometric average energy is reduced by 16%. Compared with circuit-level only optimization, the geometric average energy is reduced by 12%. Co-op Strategy tunes the learning parameter to reduce the energy of processor. Hence, the geometric average performance delay is increased by 3% compared with C-only design. But compared with S-only design, the geometric average performance delay is reduced by 3%.

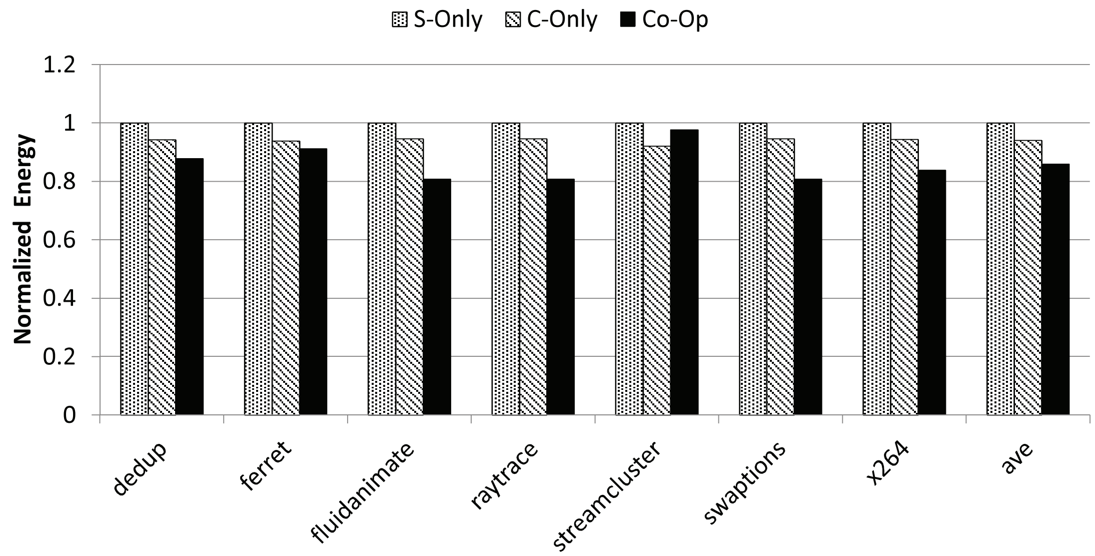
Fig. 3.17(a) and 3.17(b) respectively show the energy consumption and execution time of testing benchmarks with different strategies at 22nm technology node. S-only Strategy does not improve the reference design. The geometric average energy



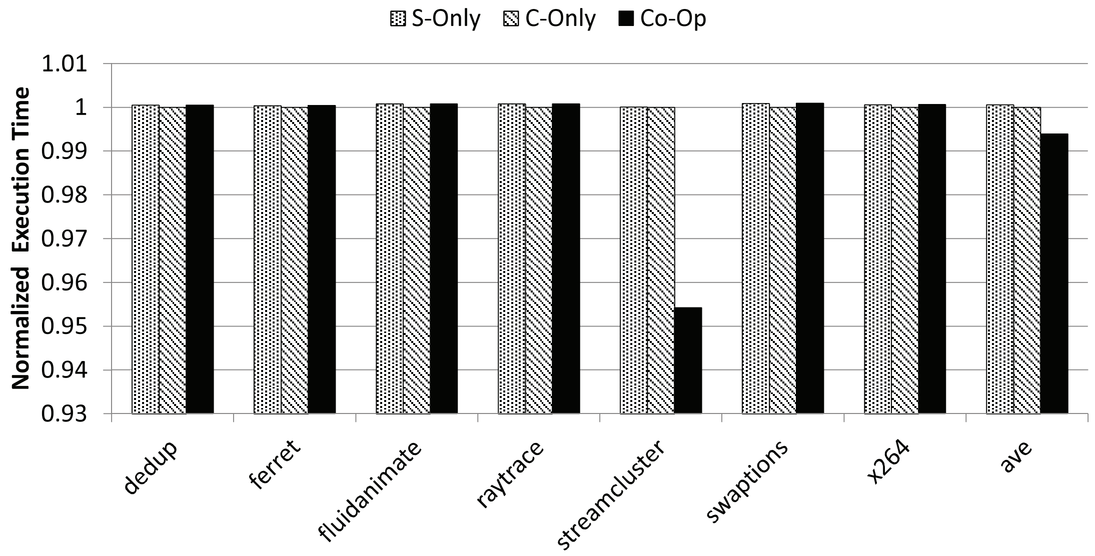
(a)

(b)

Figure 3.16: Normalized total energy consumption and execution time at 45nm technology node. The results are based on the benchmarks in the testing set. The circuit-level and system-level designs are obtained based on the training set with S-only, C-only, and Co-Op strategies respectively. The energy consumption and execution time of the three designs are normalized to the reference design. The last column shows the geometric average energy/performance of the benchmarks. (a) Normalized total energy consumption. (b) Normalized execution time.



(a)



(b)

Figure 3.17: Normalized total energy consumption and execution time at 22nm technology node. The results are based on the benchmarks in the testing set. The circuit-level and system-level designs are obtained based on the training set with S-only, C-only, and Co-Op strategies respectively. The energy consumption and execution time of the three designs are normalized to the reference design. The last column shows the geometric average energy/performance of the benchmarks. (a) Normalized total energy consumption. (b) Normalized execution time.

consumption of C-only design is reduced by 6% compared with the reference design. Compared with system-level only optimization, the geometric average energy of Co-op Strategy is reduced by 14%. Compared with circuit-level only optimization, the geometric average energy is reduced by 9%. The geometric average performance delay is almost the same as the reference design.

3.1.7 Summary

In conclusion, our proposed co-optimization strategy takes both circuit-level and system-level design issues into consideration. The results show that cross-layer co-optimization significantly reduces the total energy dissipation with small performance delay degradation. Compared with system-only or circuit-only optimizations, our proposed co-optimization have an advantage in energy saving at different technology nodes.

3.2 Design of DVFS for Multi-Core Processors

The cross-layer co-optimization strategy described in the previous section is general for DVFS system design. Nevertheless, there are special application scenarios in which additional interesting cross-layer design trade-offs exist. In this section, we discuss the DVFS for multi-core processors.

3.2.1 Background

For a multi-core processor, each core is assigned to process a workload. The workloads can be different from each other. From the system-level perspective, per-core DVFS is preferred since it generates operating point sequences for each core according to the core's workload. However, it requires that each core is supported by an isolated DC-DC converter or voltage regulators [65]. Hence, from the circuit-level perspective, per-core DVFS is not economic for modern high performance processors that are usually composed of a large amount of computation cores. A typical solution is to divide the cores into several voltage control groups [66]. The cores in each group are controlled by one DVFS controller. This power management method is named as global DVFS. The discussion in this section focuses on the design of global DVFS of multi-core processors.

3.2.2 Opportunities of Circuit/System Co-optimization

For a multi-core processor, the system-level and circuit-level design concerns and trade-offs are similar to the ones of a single-core processor. Hence, we mainly discuss the cross-layer design trade-offs.

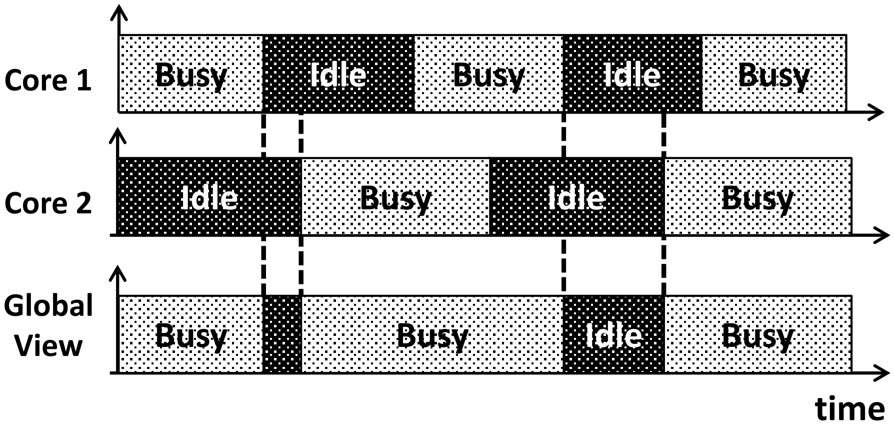


Figure 3.18: The working status of a dual-core processor.

For the global DVFS, each controller has to make decisions based on the core with the strictest performance requirement. Fig. 3.18 shows an example of a dual-core processor. The cores respectively process two workloads with different characteristics. Assume that during some time windows, one core is idle while the other one is busy. In this case, the controller would select a high-frequency operating point for both cores in order to meet the performance requirement of the busy core. The operating points with low frequencies are selected only if both cores are idle which can be very short and hard to be tracked. Hence, compared to other processors, a multi-core processor requires finer-grained DVFS to track the workloads and make delicate decisions. Shorter voltage transition time of the DC-DC converter leads to larger energy overhead that may overwhelm the energy saved by DVFS. This problem is very easy to be ignored by isolated system-level or circuit-level design strategies. Our proposed co-optimization considers the trade-offs on two levels together and thereby the optimal design can be obtained.

3.2.3 Experimental Results

In this section, we present the results of DVFS for a dual-core processor. The system-level only optimization, circuit-level only optimization and system/circuit co-optimization are compared in this section.

3.2.3.1 Experimental Setting

The setting of the experiments are tabulated in Tab. 3.4. The PDN modeling, the structure of the DC-DC converter, and the DVFS algorithm are the same as the setting of single core-processor.

3.2.3.2 Results of Co-Optimization

Fig. 3.19 shows the normalized energy consumption of benchmark *blackscholes* when two threads are processed on the dual-core processor. As can be seen, the system-level only optimization does not bring obvious improvement on energy saving. This is because that the DVFS controller makes decisions according to the condition of the busier core. As a result, the idle intervals become short and hard to be tracked. In order to implement fine-grained DVFS, the operative period must be short enough.

Table 3.4: Experimental Setting of DVFS design for Dual-Core Processor

Circuit Level		System Level	
Size of PDN	200K	Number of Cores	2
On-chip Decaps	200nf	ISA	ALPHA
Simulator	Cadence Spectre	L1 I-Cache	32KB
Optimizer	APPSPACK [52]	L1 D-Cache	32KB
Technology	90nm	L2 Cache	128KB
		Execution Model	Out of order
		Working Set	see Tab. 3.1
		Simulator	gem5 [61]
		Benchmark	PARSEC 2.1 [2]

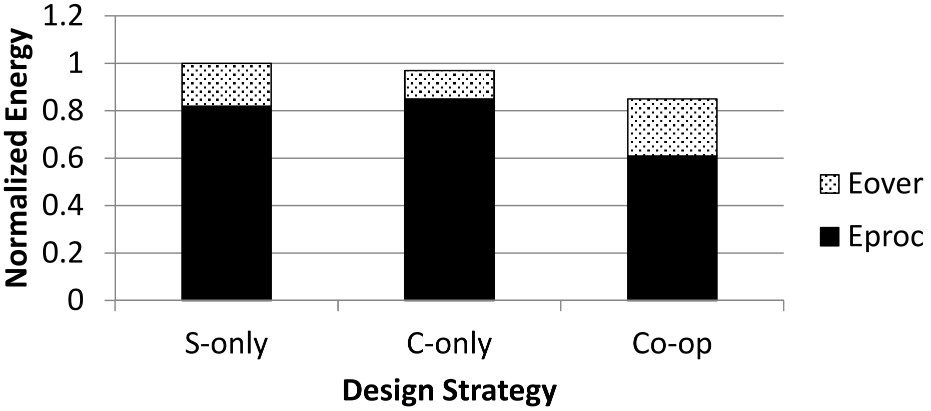


Figure 3.19: Normalized energy consumption of benchmark *blackscholes*. The results are based on the simulation with two threads processed on two cores respectively. The technology node is 90nm.

However, the minimum operative period is constrained by the DC-DC converter's transition time. Hence, the processor operates at highest frequency most of the time and thereby the benefit of the system-level only optimization is limited.

In contrast with the system-level optimization, the circuit-level optimization can shorten the transition time through circuit designs. However, the circuit has to make a trade-off between the power loss and the transition time. The power losses at certain operating points may increase. If the DVFS policy trends to select these operating points, the DVFS energy overhead E_{over} may reduce or even overwhelm the benefit of fine-grained DVFS. As a result, the improvement of energy saving is limited.

Our proposed strategy can make a trade-off between the fine-grained DVFS and the power loss of the DC-DC converter. The policy can be optimized to avoid the operating points with high power loss. Therefore, the energy can be further saved.

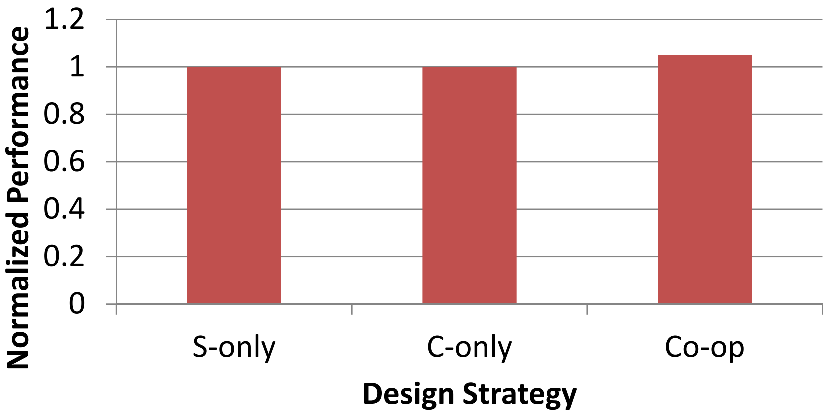
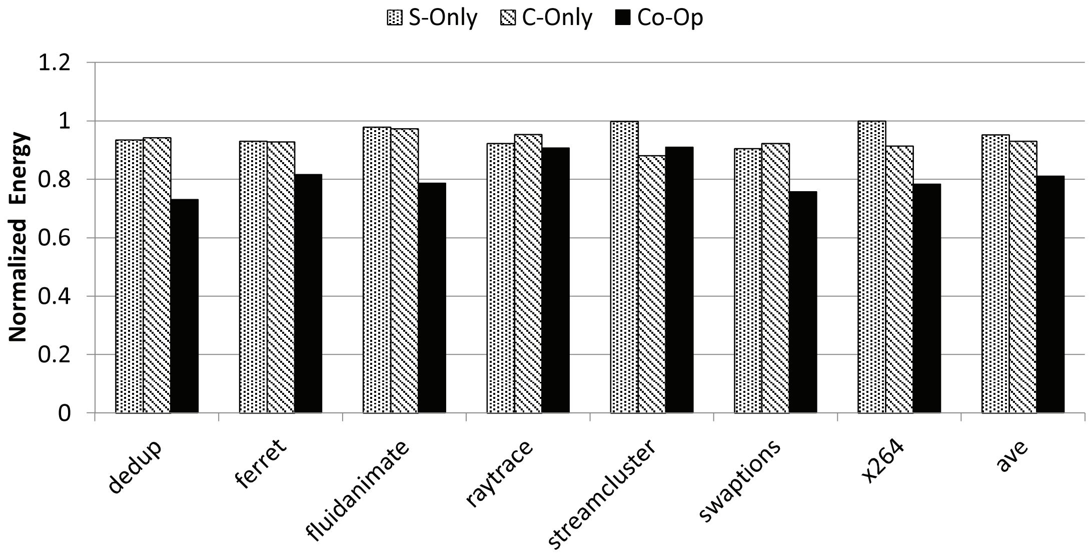


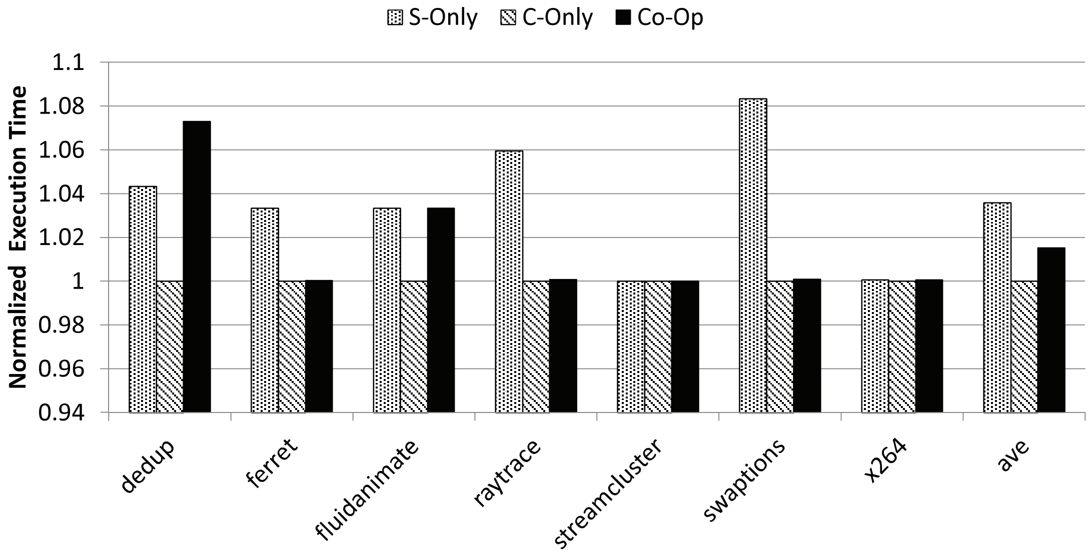
Figure 3.20: Normalized performance of benchmark *blackscholes*. The results are based on the simulation with two threads processed on two cores respectively. The technology node is 90nm.

Fig. 3.20 shows the normalized performance delay of benchmark *blackscholes* when two threads are processed on the dual-core processor. As discussed above, the processor operates at high-frequency operating points most of time with system-level only or circuit-level only optimization. Hence, the performance delay is very close to the reference design. For the proposed strategy, the energy saving mainly comes from fine-grained DVFS that is based on the trade-off between the energy and the performance delay. As a result, the performance delay is higher than the other two strategies.

Fig. 3.21(a) and 3.21(b) respectively show the energy consumption and execution time of PARSEC benchmarks in the testing set with different strategies at 90nm technology node. S-Only Strategy utilized learning policy to balance energy consumption and performance delay. However, the DVFS controller has to make decisions based on the busier core for a dual-core processor. The geometric average energy consumption is only reduce by 4% compared with the reference design. C-Only strategy uses the same DVFS controller as the reference design. Hence, DVFS operative cannot be optimized for the workloads. The geometric average energy consumption of C-Only design is reduced by 7% compared with the reference design. Co-Op Strategy optimizes the circuit-level and system-level designs together. DC-DC converter transition time and the DVFS operative time are optimized together for the workloads of dual-core processor. Hence, finer-grain DVFS is applied to the processor. Compared with system-level only optimization, the geometric average energy is reduced by 15%. Compared with circuit-level only optimization, the geometric average energy is reduced by 13%. The geometric average performance delay is increased by 1.5% compared with C-Only design. But compared with S-Only design, the geometric average performance delay is reduced by 2%.



(a)



(b)

Figure 3.21: Normalized total energy consumption and execution time of the dual-core processor at 90nm technology node. The results are based on the benchmarks in the testing set. The circuit-level and system-level designs are obtained based on the training set with S-only, C-only, and Co-Op strategies respectively. The energy consumption and execution time of the three designs are normalized to the reference design. The last column shows the geometric average energy/performance of the benchmarks. (a) Normalized total energy consumption. (b) Normalized execution time.

3.2.4 Summary

We analyze the specific trade-offs of DVFS for dual-core processor. Our proposed co-optimization flow can be utilized to optimize the global DVFS policy and the DC-DC converter together. The performance of co-optimization strategy is compared with system-only optimization and circuit-only optimization. Our proposed strategy reduces energy by 15% and 13% respectively compared with the system-only optimization and the circuit-only optimization.

3.3 Design of DVFS for Power-Gated Processors

3.3.1 Background

Power gating technique is widely used in modern processors to save leakage consumption. In this section, we discuss the DVFS for power-gated processors. For a power-gated processor, each local power domain is connected with global power delivery network through sleep transistors. When the processor is idle, sleep transistors are turned off to reduce the leakage consumption. The DVFS system design of the power-gated processor is more complex due to the interplay between DVFS and power gating.

A large body of works has been proposed to improve the performance of power-gated processor. Some works propose circuit-level solutions to address supply noise issues. Stepwise turning on of sleep transistors is used in [32] to suppress the rush current noise. The amount of rush current is controlled through slowing down the charging process. Delay skewing of sleep transistor is proposed in [33] to avoid simultaneously turning on a large amount of sleep transistors and reduce the rush current. Multiple wake-up phases are proposed in [34] to slow down the turning on procedure until the voltage of the local grid rises high enough. Multiple sleep modes with different sleep depths are proposed in [16, 35] to trade off between wake-up penalty and leakage saving.

Some works propose system-level power gating polices to avoid negative energy saving. As introduced in Chapter 2, the time during which leakage saving of power gating compensates the energy overhead is the break even time T_{BE}. The leakage saved by power gating is negative if idle time is shorter than the break even time. In other words, the energy overhead overwhelms the leakage saving. Timeout policies are proposed in [67, 68]. In a timeout policy, the processor is turned off if it is idle for more than a specified timeout period. Predictive policies are proposed in [69, 70]. In a predictive policy, the power gating decision is made as soon as the idle time arrives. Stochastic policies [38, 71]. These policies model the request arrival and device power state changes as stochastic processes.

Some existing works discuss the interplay between power and DVFS [24, 62]. However, this discussion focuses on the policy level. The cross-layer interaction can significantly influence the comprehensive performance of a power-gated processor.

3.3.2 Circuit-Level Design

Besides the DC-DC converter, the power-gated PDN is another circuit-level design component. Typical power-gated PDN designs presented in Fig. 3.4 are utilized for the discussion and analysis in this chapter.

For a power-gated PDN design, the main trade-off is between power integrity and power efficiency. The superposition of switching noise and rush current noise should be controlled under the maximum tolerable voltage drop in order to meet the power integrity requirement. Switching noise is suppressed through local decaps while rush current noise is controlled by slowing down the turn-on procedure. The turn-on time determines the efficiency of power gating. First, long turn-on time increases the delay of power gating. In addition, leakage power is consumed during the turn-on procedure and thereby energy overhead increases. More details of the design trade-offs are discussed in Chapter 2.1.2.

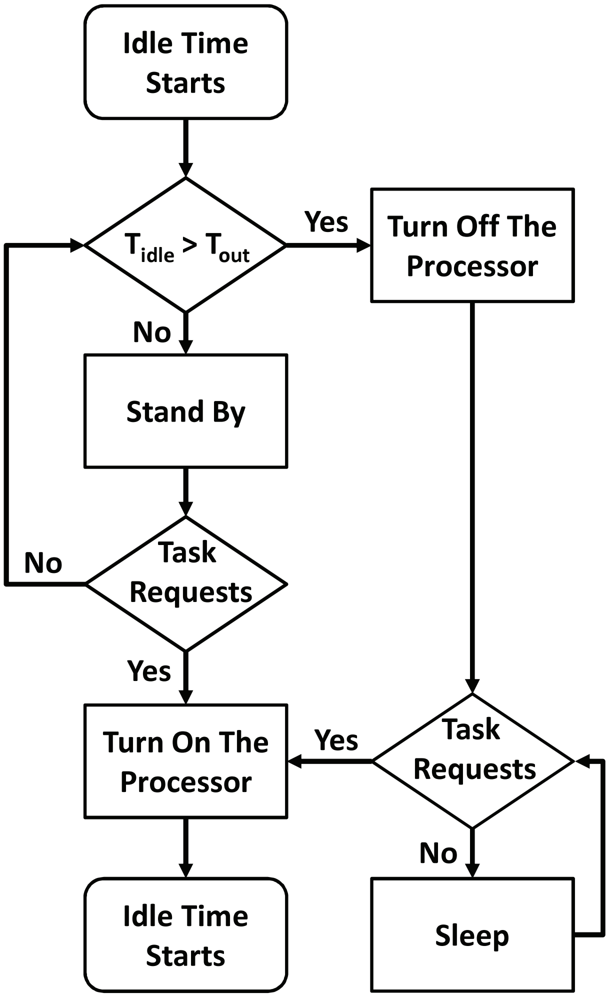


Figure 3.22: Flowchart of static timeout power gating policy. T_{idle} is the idle time. T_{out} is the timeout parameter.

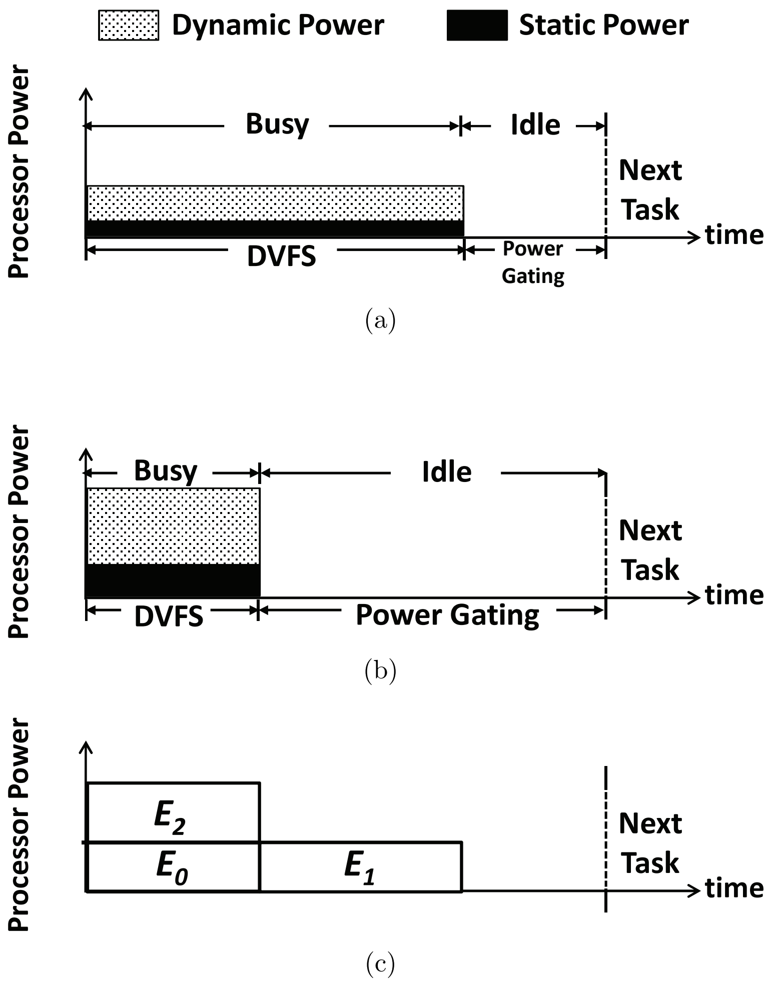
3.3.3 System-Level Design

Besides the DVFS controller, the power gating controller is another design component at the system level. In this dissertation, we employ a static timeout policy to control the power gating procedure as shown in Fig. 3.22. A static timeout policy is parameterized by the timeout parameter T_{out}. A long idle time starts after one task is completed by the processor. Power gating is not launched as soon as the processor falls idle. The processor keeps standby before $T_{idle} > T_{out}$. During this procedure, the processor keeps consuming leakage power. The processor is turned off when it remains idle for more than T_{out} time ($T_{idle} > T_{out}$). Before a new task request arrives, the processor keeps asleep for leakage saving. The processor is turned on again when a new task requests processing. Through using the static timeout power gating policy, power gating cannot be launched for short idle periods. In this case, negative energy saving of power gating is avoided to some extent.

For a static timeout power gating controller, the main trade-off is between leakage saving and energy overhead. On one hand, power gating with a small timeout parameter can reduce the leakage energy consumed during the standby state. But there exits a high risk of negative energy saving at the same time. The power gating may be launched even if the idle interval is shorter than the break even time. As a result, the net energy saved by power gating is negative. On the other hand, a large timeout parameter can reduce potential risk of negative energy saving. However, the leakage consumption during standby state increases and the opportunities of power gating may be dramatically reduced. Therefore, the power gating controller design has to balance between opportunities and risks in order to maximum energy saving.

3.3.4 Opportunities of Circuit/System Co-optimization

Fig. 3.23 shows the interplay between DVFS and power gating. After one task is completed, the processor is idle before the next task arrives. Power gating is usually applied to the idle interval to save the leakage consumption. The length of the idle interval highly depends on the DVFS policy. If the DVFS controller selects low-



(a)

(b)

(c)

Figure 3.23: Interplay between DVFS and power gating. (a) The processor operates at low-frequency operating points and the idle time for power gating is shortened. (b) The processor operates at high-frequency operating points and the idle time for power gating is extended. (c) Energy consumption comparison between (a) and (b).

frequency operating points, the execution time will be extended and thereby the idle interval will be shortened as shown in Fig. 3.23(a). In this case, the DVFS saves the dynamic energy at a cost of leakage energy growth. The total energy consumed is $E_0 + E_1$ as shown in Fig. 3.23(c). If the DVFS controller selects high-frequency operating points, the execution time will be shortened and thereby the idle interval will be extended as shown in Fig. 3.23(a). As a result, dynamic energy increases and static energy decreases. The total energy consumed is $E_0 + E_2$ as shown in Fig. 3.23(c). The optimal balance between power gating and DVFS highly depends on the ratio of static power to dynamic power. If dynamic power is dominant, dynamic power can be significantly scaling down through DVFS and thereby $E_1 < E_2$. In this case, DVFS should take up most of execution time as shown in Fig. 3.23(a). If static power is dominant, E_1 can not be reduced significantly by DVFS. In contrast, E_1 may increase with the execution time and thereby $E_1 > E_2$. In this case, power gating should be the dominant power management method as shown in Fig. 3.23(b).

Before the technology scaled down to the nanometer scale, the dynamic power was much higher than the static power. Hence, the DVFS played an dominant role in the power management as shown in Fig. 3.23(a). As the technology scales down, the static power becomes a significant portion of total chip power [5]. In this case, as shown in Fig. 3.23(b), operating at high-frequency operating points may be better for energy saving [72]. The trade-off between DVFS and power gating can be balanced through the system-level optimization. However, system-only optimization has its limitations. For example, if static power is dominant, the DVFS controller may employ high-frequency operating points in order to shorten DVFS operative time. Without circuit-level considerations, the DC-DC converter may have high power loss at high-frequency operating points and the total power may increase instead of being reduced.

Obviously, separated system-level or circuit-level design may lead to non-optimal designs. Our proposed cross-layer co-optimization method can be used to design the controller and the DC-DC converter together.

3.3.5 Experimental Results

In this section, we present the results of DVFS for a dual-core processor at different technology nodes. The system-level only optimization, circuit-level only optimization and system/circuit co-optimization are compared in this section.

3.3.5.1 Experimental Setting

The setting of the experiments are given in Tab. 3.5. The structure of the DC-DC converter is shown in Fig. 3.2(a). The DVFS policy is given by Alg. 1. The DVFS online learning controller and energy/performance calculator are implemented by C++. The circuit-level and global optimizers are implemented by a nonlinear optimization problem solver APPSPACK (Asynchronous Parallel Pattern Search Package) [52]. The training set and testing set are the same as mention in Chapter 3.1.6.1.

Table 3.5: Experimental Setting of DVFS design for Power-Gated Processor

Circuit Level		System Level	
Size of PDN	200K	Number of Cores	1
On-chip Decaps	200nf	ISA	ALPHA
Simulator	Cadence Spectre	L1 I-Cache	32KB
Optimizer	APPSPACK [52]	L1 D-Cache	32KB
Technology	90nm	L2 Cache	128KB
Turn-On Time	1000ns	Execution Model	Out of order
		Working Set	see Tab. 3.1
		Simulator	gem5 [61]
		Benchmark	PARSEC 2.1 [2]
		Timeout Parameter	1K Core Cycles[1]

[1] processor operates at maximum frequency

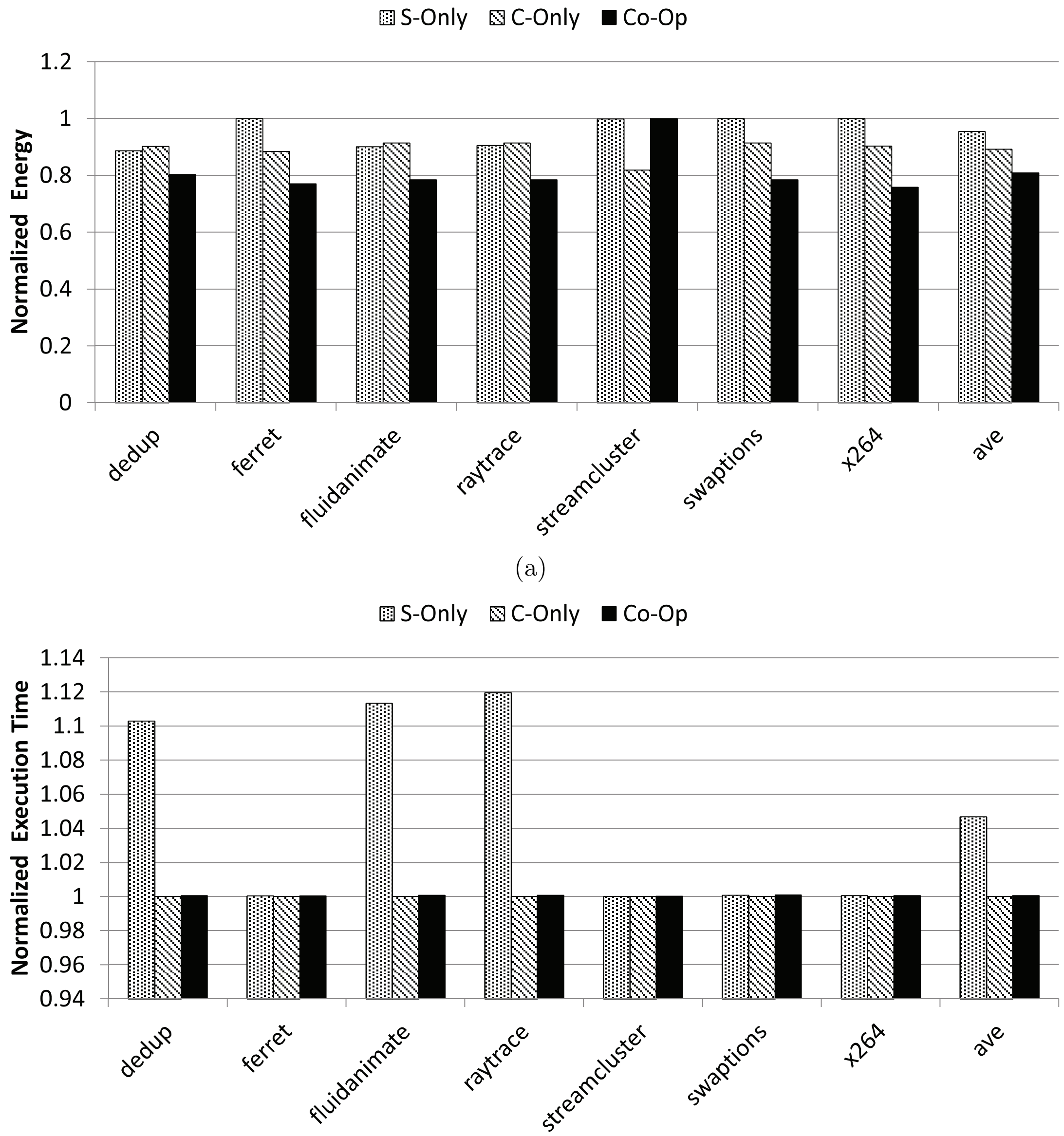
3.3.5.2 Results of Co-Optimization at 90nm Technology Node

Fig. 3.24(a) and 3.24(b) respectively show the energy consumption and execution time of SPARC benchmarks in the testing set with different strategies at 90nm technology node. Both system-level only optimization and circuit-level only optimization reduce the total energy consumptions compared with the reference design. The energy saving of the system-level only optimization mainly comes from the balance between DVFS and power gating. However, the DC-DC converter design is same as the reference design. The DC-DC converter may have a high power loss at the frequently selected operating points. Hence, the DVFS energy overhead limits the energy saving of system-level only optimization. For circuit-level only optimization, the energy is mainly saved through reducing the power loss of the DC-DC converter. However, without system-level considerations, the DC-DC converter may have high power loss at frequently selected operating points and thereby the total energy consumption my increase. Our proposed co-optimization strategy takes both system-level and circuit-level trade-offs into consideration. Compared with the system-level only optimization, the geometric average energy of the co-optimization is reduced by 15% and the performance delay is reduced by 4%. Compared with the circuit-level only optimization, the geometric average energy of the co-optimization is reduced by 9%.

3.3.5.3 Results of Co-Optimization at Advanced Technology Node

As technology scales down, static power gradually becomes the dominant component in the total energy consumption of VLSI circuits. Hence, the DVFS controller tends to select higher-frequency operating points in order to enter the sleep mode quickly.

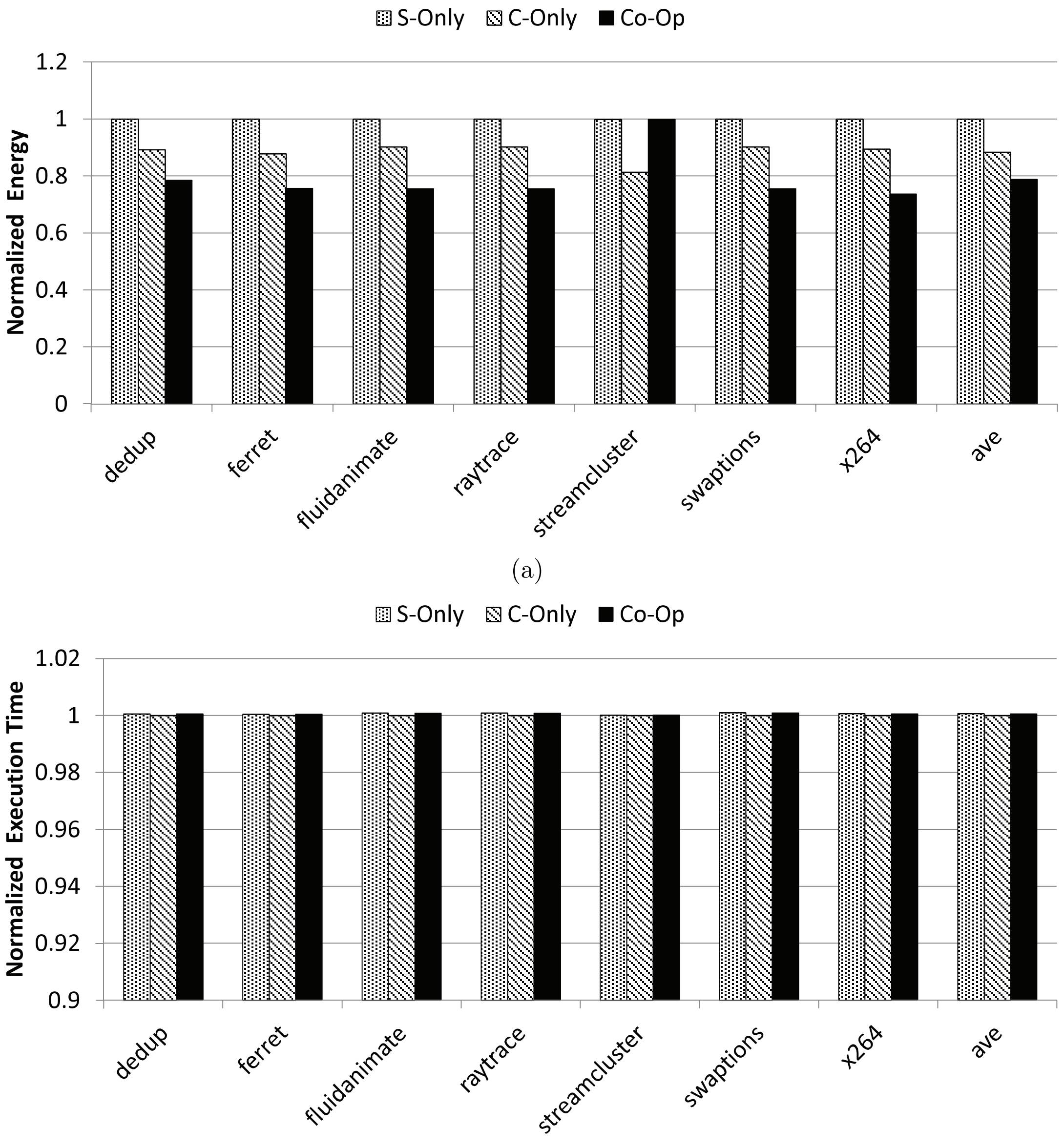
Fig. 3.25(a) and 3.25(b) respectively show the energy consumption and execu-



(a)

(b)

Figure 3.24: Normalized total energy consumption and execution time of power-gated processor at 90nm technology node. The results are based on the benchmarks in the testing set. The circuit-level and system-level designs are obtained based on the training set with S-only, C-only, and Co-Op strategies respectively. The energy consumption and execution time of the three designs are normalized to the reference design. The last column shows the geometric average energy/performance of the benchmarks. (a) Normalized total energy consumption. (b) Normalized execution time.



(a)

(b)

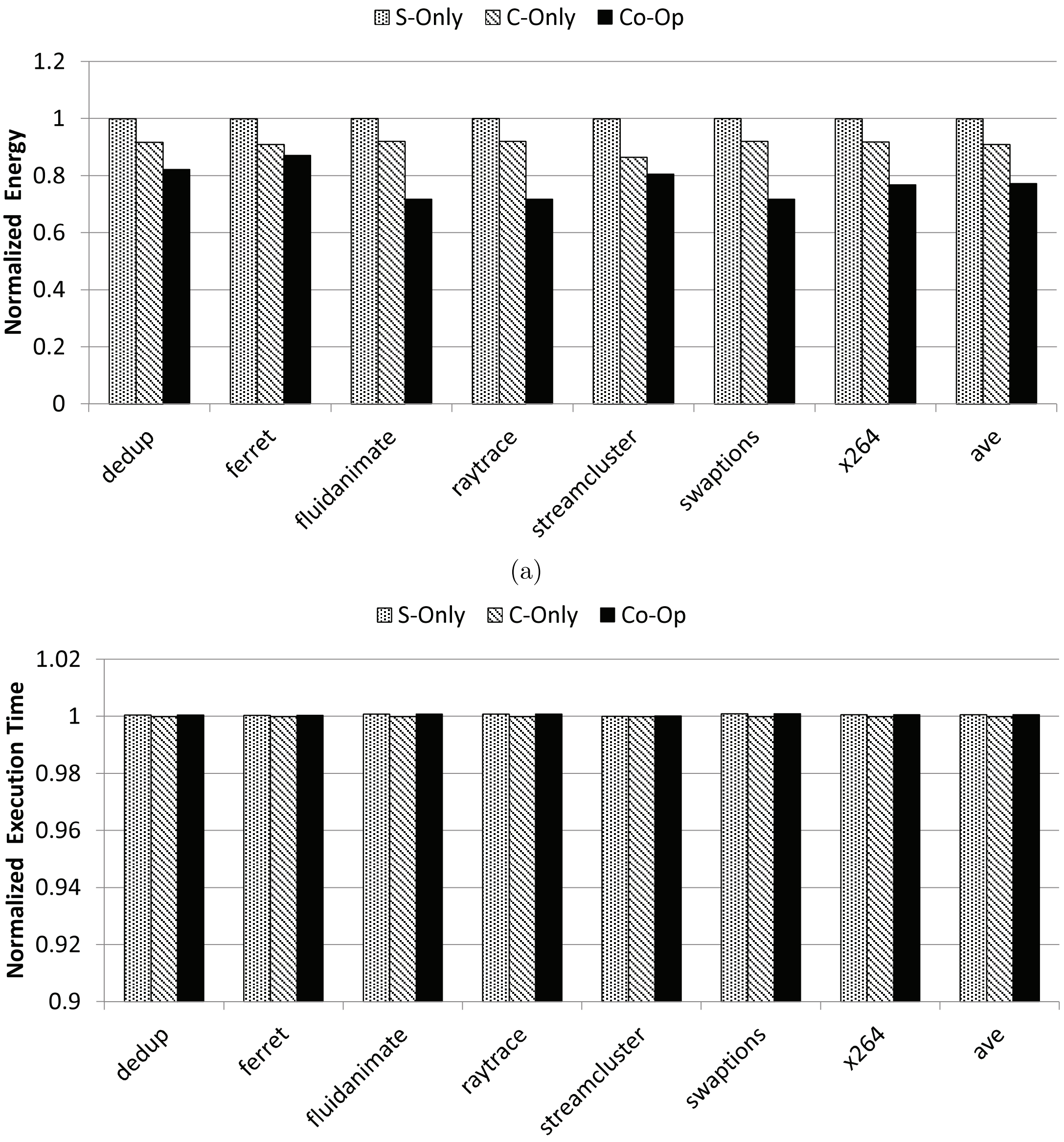
Figure 3.25: Normalized total energy consumption and execution time of power-gated processor at 45nm technology node. The results are based on the benchmarks in the testing set. The circuit-level and system-level designs are obtained based on the training set with S-only, C-only, and Co-Op strategies respectively. The energy consumption and execution time of the three designs are normalized to the reference design. The last column shows the geometric average energy/performance of the benchmarks. (a) Normalized total energy consumption. (b) Normalized execution time.

tion time of SPARC benchmarks in the testing set with different strategies at 45nm technology node. All of them tends to select high-frequency operating points. Hence, the performance delays of them are very similar to each other. For system-level only optimization, the power loss of the DC-DC converter is not considered. The large power loss at high-frequency operating point leads to additional energy consumption. For circuit-level only optimization, the DC-DC converter is optimized to minimize its largest power loss that may not appear at high-frequency operating points. As a result, the system may consume lots of energy at high-frequency operating points that are frequently selected by the DVFS controller. Our proposed co-optimization strategy takes both system-level and circuit-level trade-offs into consideration. Compared with the system-level only optimization, the geometric average energy of the co-optimization is reduced by 21%. Compared with the circuit-level only optimization, the geometric average energy of the co-optimization is reduced by 11%.

Fig. 3.26(a) and 3.26(b) respectively show the energy consumption and execution time of SPARC benchmarks in the testing set with different strategies at 22nm technology node. The performance delays of them are very similar since all of them tends to select high voltage/freqeuency operating points. Our proposed co-optimization strategy takes both system-level and circuit-level trade-offs into consideration. Compared with the system-level only optimization, the geometric average energy of the co-optimization is reduced by 23%. Compared with the circuit-level only optimization, the geometric average energy of the co-optimization is reduced by 15%.

3.3.6 Summary

We analyze the specific trade-offs of DVFS for power-gated processor. Our proposed co-optimization flow can be utilized to optimize the DVFS policy and the DC-DC converter together. The performance of co-optimization strategy is com-



(a)

(b)

Figure 3.26: Normalized total energy consumption and execution time of power-gated processor at 45nm technology node. The results are based on the benchmarks in the testing set. The circuit-level and system-level designs are obtained based on the training set with S-only, C-only, and Co-Op strategies respectively. The energy consumption and execution time of the three designs are normalized to the reference design. The last column shows the geometric average energy/performance of the benchmarks. (a) Normalized total energy consumption. (b) Normalized execution time.

pared with system-only optimization and circuit-only optimization.

4. CONCLUSIONS AND FUTURE WORK

4.1 Conclusions

This dissertation presents new design strategies and approaches to address design issues associated with power gating and DVFS.

For the power-gated PDN deigns, on-chip decoupling strategies are presented. The interactions between switching noise, rush current noise, and leakage saving are the main design challenges in power-gated PDN with single supply voltage. Global decaps are firstly proposed to suppress rush current noise. Global decaps provide part of rush current during the wake-up procedure and thereby reduce the rush current noise. With the application of global decaps, the power gating technique saves more leakage since the interaction between turn-on time and rush current noise is relaxed. However, the on-chip white space for decoupling capacitors is expensive and limited. The PDN design has to trade off between switching noise and rush current noise if the total decap budget is very tight. In this case, it is very hard to meet the requirements of power efficiency and power integrity at the same time. Re-routable decaps are proposed to address this problem. Re-routable decaps can provide different functions through two controlled switches. First, re-routable decaps can act as local decaps to suppress the switching noise when the local grid is active. Second, re-routable decaps can act as global decaps through routing to the global VDD grid when the local grid is turned off. The charges of re-routable decaps are preserved since they are connected to the global grid during the idle time. Hence, they generate smaller rush current noise than same amount of local decaps. Therefore, both switching noise and rush current noise can be suppressed by re-routable decaps. Leakage saving is significantly increased through utilization of re-routable decaps. Besides the interaction between power efficiency and power integrity, trade-off variation is another challenge for power-gated PDN with multiple supply voltages. Different decap configurations are required at different supply voltage levels. A flexible decap strategy based upon use of re-routable decaps is proposed to address varying design tradeoffs at different voltage levels. In the strategy, re-routable decaps are partitioned into two groups to generate flexible configurations. At the lower supply voltage, the re-routable decaps of one group act as normal re-routable decaps while the ones of the other group act as the global decaps to enhance the rush current noise suppression. The optimal design can be achieved at both supply voltage levels with the proposed strategy.

For the DVFS system designs, system-level and circuit-level cross-layer co-optimization strategy is presented. Separate system-level or circuit-level optimization deal with the design trade-offs at each level. The interaction between the DC-DC converter and the DVFS controller cannot be considered by system-level only or circuit-level only optimization. However, the cross-layer trade-offs significantly influences the comprehensive performance of the DVFS system. A two-step co-optimization flow is proposed to take both intra-layer and cross-layer trade-offs into consideration. In the first step, we optimize the design of the DC-DC converter for power loss, output voltage transition time, and area overhead. A pareto-optimal surface of the DC-DC converter designs is created for the next step. In the second step, system-level simulation is launched to generates a series of CPU usages based on the given DVFS operative periods. The online learning DVFS controller generates a series of operating points according to the CPU usages. Based on the operating points and the power loss of the DC-DC converter, the total energy and execution time are calculated. The global optimizer tunes circuit-level converter designs and system-level learning parameters to find the optimal DVFS policy and the optimal DC-DC converter design. The proposed design strategy is evaluated based on single-core processors, dual-core processors with global DVFS, and power-gated processors with DVFS respectively. Our study shows that the co-optimization of DVFS policies and the DC-DC converter can lead to noticeable additional energy saving without significant performance degradation.

4.2 Future Work

First, re-routable decaps have great application value for VLSI circuits with multiple power-gated domains. The percentage of chips that is idle or significantly underclocked (dark silicon) increases as the VLSI technology scales down. In conventional power-gated power delivery networks, the decap configuration cannot be changed after design procedure. In this case, a huge amount of decaps is in standby most of the time. It is a colossal waste of on-chip white space. To address this problem, re-routable decaps provide a solution for decap reuse. When a local power domain is turned off, re-routable decaps of the local domain can be routed to other domains for reuse. The white space can be effectively saved by the utilization of re-routable decaps. More design issues emerge with the decap reuse among different power domains. For example, it is a critical challenge to re-route the decaps to track the workload variations. The time-variant workloads of distinct power domains are different from each other. As a result, supply noise condition remarkably varies with time. The key point of reuse is to route the decaps to the hot spots (domains with large voltage drop). Hence, it is important to track the variation and make corresponding routing actions in real time. The balance among reuse efficiency, area overhead, and power integrity is another design challenge. The efficiency of reuse is determined by the number of domains sharing the decaps. However, long-distance re-routing may bring large voltage drop and area overhead. Therefore, the design has to trade off between these design concerns.

Second, cross-layer co-optimization can be further developed for future DVFS designs. More circuit blocks can be taken into consideration besides the DC-DC converters. For example, for a power-gated processor with DVFS, the design of entire power delivery network can be taken into the co-optimization flow. On one hand, the control circuits of sleep transistors, decap allocation, and decap budget are all circuit level design parameters that can be optimized to trade off between supply noises and leakage consumption. On the other hand, the leakage saved through power gating directly influences the DVFS policy. Hence, the power delivery network and the DVFS controller can be optimized together to improve the comprehensive performance.

Finally, cross-layer co-optimization for special DVFS applications is another future direction. For example, DVFS for devices powered by solar energy can be more complex. Solar energy is may be converted to electrical energy though a photovoltaic panel. A DC-DC converter can be then used to convert the photovoltaic panel's output voltage to the desired power supply voltages of various operating points. On one hand, solar energy may not be stable due to the outdoor environment. Hence, the power loss of the DC-DC converter varies with environment significantly. Without considering the variation of the environment and power loss, the DVFS controller may provide sub-optimal operating points for the processor. On the other hand, the energy overhead of the DC-DC converter cannot be optimized in the entire range of operating points. Hence, separate system-level optimization may also lead to sub-optimal performance without considering the DVFS policy. The energy consumption is an even more dominant concern of solar supply devices. Therefore, system-level and circuit-level co-optimization is more important for such special applications.

REFERENCES

[1] Joseph N Kozhaya, Sani R Nassif, and Farid N Najm. A multigrid-like technique for power grid analysis. *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, 21(10):1148–1160, 2002.

[2] Christian Bienia. *Benchmarking Modern Multiprocessors*. PhD thesis, Princeton University, January 2011.

[3] Michael B. Taylor. Is dark silicon useful? harnessing the four horesemen of the coming dark silicon apocalypse. In *Design Automation Conference*, 2012.

[4] Hadi Esmaeilzadeh, Emily Blem, Renee St Amant, Karthikeyan Sankaralingam, and Doug Burger. Dark silicon and the end of multicore scaling. *IEEE Micro*, 32(3):122–134, 2012.

[5] ITRS. The international technlogy roadmap for semiconductors 2013 edition, 2013.

[6] Zhigang Hu, Alper Buyuktosunoglu, Viji Srinivasan, Victor Zyuban, Hans Jacobson, and Pradip Bose. Microarchitectural techniques for power gating of execution units. In *Proceedings of the 2004 international symposium on Low power electronics and design*, pages 32–37. ACM, 2004.

[7] First the tick, now the tock: Next generation Intel microarchitecture (Nehalem). Intel Whitepaper, 2008.

[8] Jacob Leverich, Matteo Monchiero, Vanish Talwar, Parthasarathy Ranganathan, and Christos Kozyrakis. Power management of datacenter workloads using per-core power gating. *Computer Architecture Letters*, 8(2):48–51, 2009.

[9] Sin-Yu Chen, Rung-Bin Lin, Hui-Hsiang Tung, and Kuen-Wey Lin. Power gating design for standard-cell-like structured asics. In *Proceedings of the Conference on Design, Automation and Test in Europe*, pages 514–519. European Design and Automation Association, 2010.

[10] Intel. Mobile 4th gen intel core processor family: Datasheet, vol. 1. Intel Whitepaper, 2013.

[11] Meeta S Gupta, Jarod L Oatley, Russ Joseph, Gu-Yeon Wei, and David M Brooks. Understanding voltage variations in chip multiprocessors using a distributed power-delivery network. In *Design, Automation & Test in Europe Conference & Exhibition, 2007. DATE'07*, pages 1–6. IEEE, 2007.

[12] Haihua Su, Sachin S Sapatnekar, and Sani R Nassif. Optimal decoupling capacitor sizing and placement for standard-cell layout designs. *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, 22(4):428–436, 2003.

[13] Shiyou Zhao, Kaushik Roy, and Cheng-Kok Koh. Decoupling capacitance allocation and its application to power-supply noise-aware floorplanning. *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, 21(1):81–92, 2002.

[14] Hailin Jiang, Malgorzata Marek-Sadowska, and Sani R Nassif. Benefits and costs of power-gating technique. In *Computer Design: VLSI in Computers and Processors, 2005. ICCD 2005. Proceedings. 2005 IEEE International Conference on*, pages 559–566. IEEE, 2005.

[15] Suhwan Kim, Stephen V Kosonocky, and Daniel R Knebel. Understanding and minimizing ground bounce during mode transition of power gating structures.

In *Proceedings of the 2003 international symposium on Low power electronics and design*, pages 22–25. ACM, 2003.

[16] Kanak Agarwal, Kevin Nowka, Harmander Deogun, and Dennis Sylvester. Power gating with multiple sleep modes. In *Proceedings of the 7th International Symposium on Quality Electronic Design*, pages 633–637. IEEE Computer Society, 2006.

[17] Ken-ichi Kawasaki, Tetsuyoshi Shiota, Koichi Nakayama, and Atsuki Inoue. A sub-μs wake-up time power gating technique with bypass power line for rush current support. In *VLSI Circuits, 2008 IEEE Symposium on*, pages 146–147. IEEE, 2008.

[18] Shi-Hao Chen and Jiing-Yuan Lin. Implementation and verification practices of DVFS and power gating. In *Proc. Int. Symp. VLSI Design, Automation and Test VLSI-DAT '09*, pages 19–22, 2009.

[19] Tong Xu and Peng Li. Design and optimization of power gating for DVFS applications. In *2012 13th International Symposium on Quality Electronic Design*, pages 391–397, Mar. 2012.

[20] Simcha Gochman, Avi Mendelson, Alon Naveh, and Efraim Rotem. Introduction to intel core duo processor architecture. *Intel Technology Journal*, 10(2):89–97, 2006.

[21] Rinkle Jain and Seth Sanders. A 200ma switched capacitor voltage regulator on 32nm cmos and regulation schemes to enable dvfs. In *Power Electronics and Applications (EPE 2011), Proceedings of the 2011-14th European Conference on*, pages 1–10. IEEE, 2011.

[22] Wonyoung Kim, D. Brooks, and Gu-Yeon Wei. A fully-integrated 3-level DC-DC converter for nanosecond-scale DVFS. *IEEE Journal of Solid-State Circuits*, 47(1):206–219, Jan. 2012.

[23] Khurram Bhatti, Cecile Belleudy, and Michel Auguin. Power management in real time embedded systems through online and adaptive interplay of dpm and dvfs policies. In *Embedded and Ubiquitous Computing (EUC), 2010 IEEE/IFIP 8th International Conference on*, pages 184–191. IEEE, 2010.

[24] Guaurav Dhiman and Tajana Simunic Rosing. Dynamic power management using machine learning. In *ICCAD*, pages 747–754, Nov. 2006.

[25] Kihwan Choi, Ramakrishna Soma, and Massoud Pedram. Fine-grained dynamic voltage and frequency scaling for precise energy and performance tradeoff based on the ratio of off-chip access to on-chip computation times. *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, 24(1):18–28, 2005.

[26] Krishna K Rangan, Gu-Yeon Wei, and David Brooks. Thread motion: fine-grained power management for multi-core systems. In *ACM SIGARCH Computer Architecture News*, volume 37, pages 302–313. ACM, 2009.

[27] Jong Sung Lee, Kevin Skadron, and Sung Woo Chung. Predictive temperature-aware dvfs. *Computers, IEEE Transactions on*, 59(1):127–133, 2010.

[28] Ryan Cochran, Can Hankendi, Ayse K Coskun, and Sherief Reda. Pack & cap: adaptive dvfs and thread packing under power caps. In *Proceedings of the 44th annual IEEE/ACM international symposium on microarchitecture*, pages 175–185. ACM, 2011.

[29] Ankush Varma, Brinda Ganesh, Mainak Sen, Suchismita Roy Choudhury, Lakshmi Srinivasan, and Bruce Jacob. A control-theoretic approach to dynamic voltage scheduling. In *Proceedings of the 2003 international conference on Compilers, architecture and synthesis for embedded systems*, pages 255–266. ACM, 2003.

[30] Ravindra Jejurikar, Cristiano Pereira, and Rajesh Gupta. Leakage aware dynamic voltage scaling for real-time embedded systems. In *Proceedings of the 41st annual Design Automation Conference*, pages 275–280. ACM, 2004.

[31] Ana Azevedo, Ilya Issenin, Radu Cornea, Rajesh Gupta, Nikil Dutt, Alex Veidenbaum, and Alexandru Nicolau. Profile-based dynamic voltage scheduling using program checkpoints. In *Proceedings of the conference on Design, automation and test in Europe*, page 168. IEEE Computer Society, 2002.

[32] Suhwan Kim, Chang Jun Choi, Deog-Kyoon Jeong, Stephen V Kosonocky, and Sung Bae Park. Reducing ground-bounce noise and stabilizing the data-retention voltage of power-gating structures. *Electron Devices, IEEE Transactions on*, 55(1):197–205, 2008.

[33] Kimiyoshi Usami, Toshiaki Shirai, Tatsunori Hashida, Hiroki Masuda, Seidai Takeda, Mitsutaka Nakata, Naomi Seki, Hideharu Amano, Mitaro Namiki, Masashi Imai, et al. Design and implementation of fine-grain power gating with ground bounce suppression. In *VLSI Design, 2009 22nd International Conference on*, pages 381–386. IEEE, 2009.

[34] Rahul Singh, Jong-Kwan Woo, Hyunjoong Lee, So Young Kim, and Suhwan Kim. Power-gating noise minimization by three-step wake-up partitioning. *Circuits and Systems I: Regular Papers, IEEE Transactions on*, 59(4):749–762, 2012.

[35] Harmander Singh, Kanak Agarwal, Dennis Sylvester, and Kevin J Nowka. Enhanced leakage reduction techniques using intermediate strength power gating. *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, 15(11):1215–1224, 2007.

[36] Tong Xu, Peng Li, and Boyuan Yan. Decoupling for power gating: Sources of power noise and design strategies. In *Proc. 48th ACM/EDAC/IEEE Design Automation Conf. (DAC)*, pages 1002–1007, 2011.

[37] Kyung Ki Kim, Haiqing Nan, and Ken Choi. Power gating for ultra-low voltage nanometer ICs. In *Proc. IEEE Int Circuits and Systems (ISCAS) Symp*, pages 1472–1475, 2010.

[38] Tajana Simunic, Luca Benini, Peter Glynn, and Giovanni De Micheli. Event-driven power management. *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, 20(7):840–857, 2001.

[39] Zhiyuan Ren, Bruce H Krogh, and Radu Marculescu. Hierarchical adaptive dynamic power management. *Computers, IEEE Transactions on*, 54(4):409–420, 2005.

[40] Yanzhi Wang, Qing Xie, Ahmed Ammari, and Massoud Pedram. Deriving a near-optimal power management policy using model-free reinforcement learning and bayesian classification. In *Proceedings of the 48th Design Automation Conference*, pages 41–46. ACM, 2011.

[41] Inshad Chowdhury and Dongsheng Ma. Design of reconfigurable and robust integrated sc power converter for self-powered energy-efficient devices. *Industrial Electronics, IEEE Transactions on*, 56(10):4018–4028, 2009.

[42] Hanh-Phuc Le, Michael Seeman, Seth R. Sanders, Visvesh S. Sathe, Samuel Naffziger, and Elad Alon. A 32nm fully integrated reconfigurable switched-capacitor dc-dc converter delivering $0.55w/mm^2$ at 81% efficiency. In *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2010 IEEE International*, pages 210–211, Feb 2010.

[43] Yogesh Ramadass, Ayman Fayed, Baher Haroun, and Anantha Chandrakasan. A $0.16mm^2$ completely on-chip switched-capacitor dc-dc converter using digital capacitance modulation for ldo replacement in 45nm cmos. In *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2010 IEEE International*, pages 208–209, Feb 2010.

[44] Yogesh K Ramadass, Ayman A Fayed, and Anantha P Chandrakasan. A fully-integrated switched-capacitor step-down dc-dc converter with digital capacitance modulation in 45 nm cmos. *Solid-State Circuits, IEEE Journal of*, 45(12):2557–2565, 2010.

[45] Pingqiang Zhou, Dong Jiao, Chris H Kim, and Sachin S Sapatnekar. Exploration of on-chip switched-capacitor dc-dc converter for multicore processors using a distributed power delivery network. In *CICC*, pages 1–4, 2011.

[46] Adel Deris Zadeh, Ehsan Adib, and Hosein Farzanehfard. A new 3-level converter for switched reluctance motor drive. In *Electrical Engineering (ICEE), 2011 19th Iranian Conference on*, pages 1–6. IEEE, 2011.

[47] Rami A Abdallah, Pradeep S Shenoy, Naresh R Shanbhag, and Philip T Krein. System energy minimization via joint optimization of the dc-dc converter and the core. In *Proceedings of the 17th IEEE/ACM international symposium on Low-power electronics and design*, pages 97–102. IEEE Press, 2011.

[48] Yongseok Choi, Naehyuck Chang, and Taewhan Kim. DC-DC converter-aware power management for low-power embedded systems. *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, 26(8):1367–1381, Aug.

[49] Chengwen Pei, Roger Booth, Herbert Ho, Naoyoshi Kusaba, Xi Li, M-J Brodsky, Paul Parries, Hulling Shang, Rama Divakaruni, and Subramanian Iyer. A novel, low-cost deep trench decoupling capacitor for high-performance, low-power bulk cmos applications. In *Solid-State and Integrated-Circuit Technology, 2008. ICSICT 2008. 9th International Conference on*, pages 1146–1149. IEEE, 2008.

[50] Bardia Bozorgzadeh and Ali Afzali-Kusha. Novel mos decoupling capacitor optimization technique for nanotechnologies. In *VLSI Design, 2009 22nd International Conference on*, pages 175–180. IEEE, 2009.

[51] Sani R Nassif. Power grid analysis benchmarks. In *Proceedings of the 2008 Asia and South Pacific Design Automation Conference*, pages 376–381. IEEE Computer Society Press, 2008.

[52] Joshua D Griffin, Tamara G Kolda, and Robert Michael Lewis. Asynchronous parallel generating set search for linearly constrained optimization. *SIAM Journal on Scientific Computing*, 30(4):1892–1924, 2008.

[53] Tejaswini Kolpe, Antonia Zhai, and Sachin S Sapatnekar. Enabling improved power management in multicore processors through clustered dvfs. In *Design, Automation & Test in Europe Conference & Exhibition (DATE), 2011*, pages 1–6. IEEE, 2011.

[54] Jacob Leverich, Matteo Monchiero, Vanish Talwar, Parthasarathy Ranganathan, and Christos Kozyrakis. Power management of datacenter workloads

using per-core power gating. *Computer Architecture Letters*, 8(2):48–51, 2009.

[55] Hyung-Ock Kim, Bong Hyun Lee, Jong-Tae Kim, Jung Yun Choi, Kyu-Myung Choi, and Youngsoo Shin. Supply switching with ground collapse for low-leakage register files in 65-nm CMOS. 18(3):505–509, 2010.

[56] B Calhoun and A Chandrakasan. Standby voltage scaling for reduced power. In *Custom Integrated Circuits Conference, 2003. Proceedings of the IEEE 2003*, pages 639–642. IEEE, 2003.

[57] Cédric Lichtenau, Mathew I. Ringler, Thomas Pflüger, Steve Geissler, Rolf Hilgendorf, Jay Heaslip, Ulrich Weiss, Peter Sandon, Norman Rohrer, Erwin Cohen, and Miles Canada. Powertune: advanced frequency and power scaling on 64b PowerPC microprocessor. In *IEEE International Solid-State Circuits Conference*, pages 356–357, Feb. 2004.

[58] Robert W Erickson and Dragan Maksimovic. *Fundamentals of power electronics*. Springer, 2001.

[59] Cahit Gezgin. Predicting load transient response of output voltage in dc-dc converters. In *Applied Power Electronics Conference and Exposition, 2004. APEC'04. Nineteenth Annual IEEE*, volume 2, pages 1339–1344. IEEE, 2004.

[60] Jaehyun Park, Donghwa Shin, Naehyuck Chang, and Massoud Pedram. Accurate modeling and calculation of delay and energy overheads of dynamic voltage scaling in modern high-performance microprocessors. In *Proceedings of the 16th ACM/IEEE international symposium on Low power electronics and design*, pages 419–424. ACM, 2010.

[61] Nathan L. Binkert, Ronald G. Dreslinski, Lisa R. Hsu, Kevin T. Lim, Ali G. Saidi, and Steven K. Reinhardt. The m5 simulator: Modeling networked sys-

tems. *IEEE Micro*, 26(4):52–60, July 2006.

[62] Gaurav Dhiman and Tajana Simunic Rosing. System-level power management using online learning. *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, 28(5):676–689, 2009.

[63] The international technlogy roadmap for semiconductors (ITRS) 2004 edition. http://public.itrs.net/, 2004.

[64] The international technlogy roadmap for semiconductors (ITRS) 2009 edition. http://public.itrs.net/, 2009.

[65] Wonyoung Kim, Meeta Sharma Gupta, Gu-Yeon Wei, and David Brooks. System level analysis of fast, per-core dvfs using on-chip switching regulators. In *High Performance Computer Architecture, 2008. HPCA 2008. IEEE 14th International Symposium on*, pages 123–134. IEEE, 2008.

[66] Jason Howard, Saurabh Dighe, Yatin Hoskote, Sriram Vangal, David Finan, Gregory Ruhl, David Jenkins, Howard Wilson, Nitin Borkar, Gerhard Schrom, et al. A 48-core ia-32 message-passing processor with dvfs in 45nm cmos. In *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2010 IEEE International*, pages 108–109. IEEE, 2010.

[67] Anna R Karlin, Mark S Manasse, Lyle A McGeoch, and Susan Owicki. Competitive randomized algorithms for nonuniform problems. *Algorithmica*, 11(6):542–571, 1994.

[68] Branislav Kveton, Prashant Gandhi, Georgios Theocharous, Shie Mannor, Barbara Rosario, and Nilesh Shah. Adaptive timeout policies for fast fine-grained power management. In *PROCEEDINGS OF THE NATIONAL CONFERENCE*

ON ARTIFICIAL INTELLIGENCE, volume 22, page 1795. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2007.

[69] Mani B Srivastava, Anantha P Chandrakasan, and Robert W Brodersen. Predictive system shutdown and other architectural techniques for energy efficient programmable computation. *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, 4(1):42–55, 1996.

[70] Chi-Hong Hwang and Allen C-H Wu. A predictive system shutdown method for energy saving of event-driven computation. *ACM Transactions on Design Automation of Electronic Systems (TODAES)*, 5(2):226–241, 2000.

[71] Eui-Young Chung, Luca Benini, Alessandro Bogliolo, Yung-Hsiang Lu, and Giovanni De Micheli. Dynamic power management for nonstationary service requests. *Computers, IEEE Transactions on*, 51(11):1345–1361, 2002.

[72] Etienne Le Sueur and Gernot Heiser. Slow down or sleep, that is the question. In *USENIX Annual Technical Conference*, 2011.