

# Optimizing the Energy Efficiency of Switched-Capacitor Converters in Multiprocessor System-on-Chips with a Preset DVFS Policy

Linfeng Zheng<sup>\*†‡</sup> and Pingqiang Zhou<sup>\*</sup>

<sup>\*</sup>ShanghaiTech University, Shanghai, China

<sup>†</sup>Shanghai Institute of Microsystem and Information Technology, Chinese Academy of Sciences, Shanghai, China

<sup>‡</sup>University of Chinese Academy of Sciences, Beijing, China

**Abstract**—Some multiprocessor system-on-chips (MPSoCs) provide designers with workload information in early chip planning stage to optimize the chip’s performance. In our work, we focus on the energy efficiency of switched-capacitor converters (SCCs) in MPSoCs to efficiently support dynamic voltage and frequency scaling (DVFS). Two things motivate us to propose our design flow in MPSoCs, i.e. 1) Recent work allocates Metal-Insulator-Metal (MIM) capacitance and selects a converter ratio for each SCC to reach a higher energy efficiency. However, it brings the high overhead of its hardware implementation for various voltage and current demands. 2) DVFS policy has an unbalance distribution of its scaling decisions, which further results in an unbalance distribution of the voltage and current demands. Based on these, in our design flow, the hotspot recognition and the look-up table techniques help us to overcome the shortcomings of the recent work and harvest the benefits of the recent work. The experiment explores the different hotspot capacities’ effects on energy efficiency and overhead.

**Index Terms**—multiprocessor system-on-chips (MPSoCs), switched-capacitor converters (SCCs), energy efficiency.

## I. INTRODUCTION

Multicore chips have been widely used in edge computing and embedded systems currently [1], especially in networking, communications, signal processing and multimedia [2]. Normally, multiprocessor system-on-chips (MPSoCs) run repetitive certain tasks. They have an application-specific feature. Thus, some of MPSoCs can provide designers with workload information in early chip planning stage for optimizing power, performance and area [2], [3].

With the parallel computing techniques in MPSoCs, the variable usage of cores leaves opportunities for power management [4], which is a focal point in edge computing. The basic rule is to turn some cores off or scale down their voltage and frequency when the workload is not burdensome. One related technique is dynamic voltage and frequency scaling (DVFS).

To support voltage scaling in MPSoCs, the whole power supply system is organized with several voltage domains, especially in a heterogeneous chip. Each domain is supplied with energy by an individual voltage regulator (VR) [5]. They can scale the voltage to several levels. Among various types of VRs, switched-capacitor converters (SCCs) have drawn much attention from both industry and academia for their high energy efficiency and easy on-chip integration [6], [7]. Even so, the SCCs have large overhead on delivering energy. They consume about 10%-40% energy of the chip (see Fig. 1). Thus, improving energy conversion efficiency of SCCs is vital.

Recent work [7] has proposed a mixed integer nonlinear programming (MINLP) model to optimize power loss of SCCs

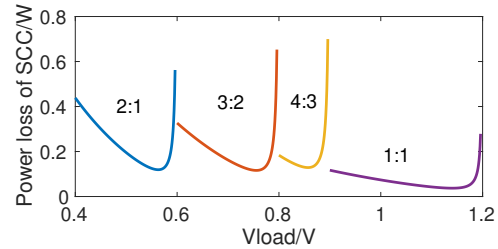


Fig. 1. A typical power loss curve of SCCs [6] with load power of 1W and  $V_{dd} = 1.2V$ . There’re four topological structures that have different ratios used to cover the large load voltage range. Each one covers a certain range of  $V_{load}$ .

with two types of variables, i.e. the allocated metal-insulator-metal (MIM) capacitance and the selected converter ratio for each SCC. Meanwhile, the traditional method which has been widely used allocates MIM capacitance to each SCC according to the area of corresponding cores and uses one certain topological structure of SCCs to cover a certain range of load voltage (see Fig. 1). Unlike the traditional method that can easily be implemented, work [7] further improves the energy efficiency but leads to the following problems:

- The model needs *the real time voltage and current for each core (see Section II-B) which we define as the demands of load circuitries*. However, solving MINLP takes hundreds of seconds [7], which is impractical to solve it in practical use of MPSoCs. A workaround is using a look-up table to record the predetermined optimized solutions.
- Considering the various voltage domains, the total number of different demands is significantly large. For example, suppose we have  $N$  independent voltage domains,  $M$  voltage levels and  $T$  intervals which cover the current distribution, we will have  $(M * T)^N$  different demands in load circuitries. It will result in a large capacity of the look-up table.
- The control circuit of converting allocated capacitance will be much complex considering the various optimized solutions.

In short, we summarize the traditional method, work [7] and the objectives of our work in Table I.

TABLE I  
THE COMPARISON BETWEEN THE TRADITIONAL METHOD AND WORK [7].

	The Traditional Method	Work [7]	Our Work
Energy Efficiency	Normal	High	High
Hardware Implementation	Easy	Hard	Easy
Implementation Overhead	Low	High	Low

Another observation is that the DVFS policy [8] which optimizes the energy delay product has an unbalance distribution on its scaling decisions. For the task in one time interval that is cpu-bound, the policy tends to use a higher voltage level. Meanwhile, the policy tends to use a lower voltage level when the task is IO-bound. Thus, the maximum and minimum voltage levels have more chances to be demanded [9]. The unbalance distribution under a certain group of repetitive tasks will further result in hotspots (see Section II-A). *The hotspots are defined as the frequent demanded voltage and current of the load circuitries.* Due to the application-specific feature of MPSoCs, some designers are able to recognize the hotspots during the design flow.

Based on these observations, we are motivated to combine the traditional method and work [7] for SCCs design in MPSoCs. To reduce the complexity and overhead of the implementation, we recognize the hotspots in the design flow. The objective of the recognition is using a few optimized hotspots to cover the DVFS intervals as many as possible in practical use. To reach the high energy efficiency, we use work [7] to optimize the hotspots. These optimized solutions will be cached in a look-up table. Our work also explores the effects of different capacities of hotspots on energy efficiency and overhead. In Section II, we elaborate our methodology in detail. The experimental results are shown in Section III.

Our contributions in this work are listed as follows:

- We propose a design flow of MPSoCs based on a rough design of SCCs to further improve their energy efficiency.
- We propose the hotspot recognition and the look-up table techniques to overcome work [7]’s shortcomings and combine the benefits of work [7] and the traditional method.
- We conduct experiments by using the GEM5 [10] simulator to follow our design flow. The experiment explores the different hotspot capacities’ effects on energy efficiency and overhead.

## II. METHODOLOGY

In our method, we improve the energy efficiency of SCCs based on SCCs using MIM capacitance. Some predetermined parameters of SCCs should be known before our design flow (see Section III-A). *The method applies to a power grid design in an application-specific MPSoC,* which indicates that the design of load circuitries, the workload and the DVFS policy are known during the procedure. We propose our design flow as follows and harvest the benefits by two key techniques. The hotspot recognition is used to use a few hotspots to cover large enough DVFS intervals, and the look-up table is used to cached optimized solutions of MINLP to realize real time optimization.

As shown in Fig. 2, the flow has 4 steps. In step 1, we use the system level simulation with a certain workload and the DVFS policy to obtain the voltage and current demands at the moment when the DVFS policy makes scaling decisions. In step 2, hotspots are recognized from the whole demands (see Section II-A). After that, the Optimizer() (see Section II-B) is used to obtain the allocated MIM capacitance and selected topological structures for each hotspot demand in step 3. In step 4, the optimized solutions are cached in a look-up table (see Section II-C).

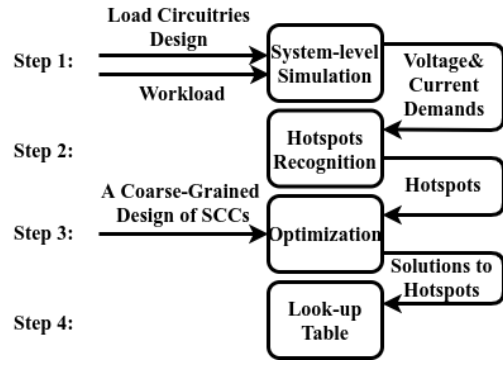


Fig. 2. The design flow.

### A. Hotspot Recognition

With the voltage and current demands obtained after step 1, we analyze the hotspots in a statistical way. Firstly, we discretize all voltage and current data. After that, we calculate the occurrence for each discretized demand. Finally, we draw a cumulative distribution function (CDF) curve and analyze the hotspots according to inflection points.

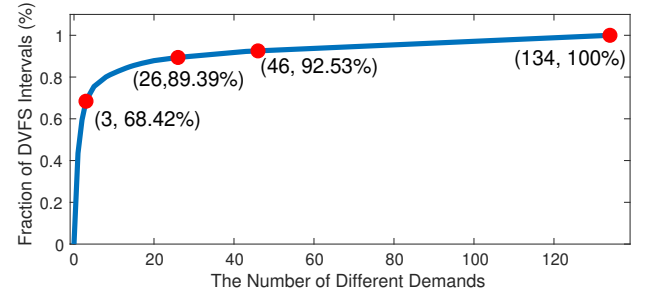


Fig. 3. The CDF curve.

The data are generated from the GEM5 simulator [10] with the SPLASH-2 benchmark suite [11]. We sort the occurrences of different demands and accumulate each one from the largest to the smallest, which generates Fig. 3. What’s more, we marked 4 possible inflection points on the curve. They help us to explore the different hotspot capacities’ effects on energy efficiency and overhead. One certain point indicates that the left part has more chances to be demanded in real use which we can regard as the hotspots. Different inflection points result in different effects on energy efficiency and implementation overhead. After this step, one of these points should be selected so that its left part can be treated as hotspots.

### B. Optimization

To further improve the energy efficiency, work [7] found that using its MINLP model shown in the follows could achieve 9%-13% improvement:

$$\begin{bmatrix} x_1^1 & \cdots & x_N^1 \\ \vdots & \ddots & \vdots \\ x_1^M & \cdots & x_N^M \end{bmatrix}, \begin{bmatrix} C_{sw}^1 \\ \vdots \\ C_{sw}^M \end{bmatrix} := \text{Optimizer} \left( \begin{bmatrix} I_{out}^1 \\ \vdots \\ I_{out}^M \end{bmatrix}, \begin{bmatrix} V_{out}^1 \\ \vdots \\ V_{out}^M \end{bmatrix} \right) \quad (1)$$

Note that  $m$  is the index of total  $M$  independent SCCs, and  $n$  is the index of total  $N$  ratios of SCCs. There’re  $M \times N$  binary variables  $x_n^m$  which indicate the  $m$ -th SCC selects the  $n$ -th ratio if  $x_n^m = 1$ .  $C_{sw}^m$  means the allocated MIM capacitance that the  $m$ -th SCC has.  $I_{out}^m$  and  $V_{out}^m$  are the current and

voltage demands of load circuitries. The other parameters in the `Optimizer()` can be predetermined as constants for a coarse-grained design. In our method, the `Optimizer()` is used for hotspot demands, i.e. the frequent demanded voltage and current. The optimized results will guide us for the further design, i.e. the topological structures and the allocated MIM capacitance for each SCC.

### C. The Look-up Table

For the look-up table design, we refer to the structure of fully associate cache [12]. The tags here are the discretized indices of voltage and current demands, and the data are the correspondingly optimized results. If one of tags is hit, the chip gets the control info for each SCC. If not, the load demands are not included in the table so that we control SCCs in the traditional way.

## III. EXPERIMENTAL RESULTS

### A. Experimental Settings

In the experiment, we use GEM5 [10] and McPAT [13] to obtain the cores' voltage and current demands with the DVFS policy [8]. The DVFS policy makes the scaling decision every  $10^5$  ns. The configuration of GEM5 is shown in Table II. Besides, according to the core's configuration and work [6], [7], we give predetermined parameters of SCCs (the constant parameters in the MINLP model) as Table III and IV shown.

TABLE II  
GEM5 CONFIGURATION.

Number of cores	4	Number of voltage levels	6
L1 data cache	32kBytes	L1 instruction cache	32kBytes
L2 cache	2MBytes	Instruction-set architecture	Alpha
Main memory	1024MBytes	Default frequency	2.50GHz
Default voltage	1.0V	Linux kernel version	2.6

TABLE III  
PREDETERMINED PARAMETERS OF SCCS.

Param.	$V_{dd}$	$N_{phase}$	$f_{sw}$	$C_{gate}$	$R_{on}$	$\sigma_{SCC}$
Value	1.2V	32	200MHz	3fF/ $\mu$ m	130 $\Omega \cdot \mu$ m	512 $\mu$ l( $\mu$ F·MHz)

TABLE IV  
TOPOLOGY-DEPENDENT PARAMETERS OF SCCS.

Conversion Ratio	$V_{nl}$	$N_{sw}$	$M_{topo}$	$M_{sw}$	$\gamma_{SCC}$
2:1	0.6V	4	2	2	2
3:2	0.8V	7	9/8	2	1
4:3	0.9V	10	8/9	7/3	2/3
1:1	1.2V	2	1/2	1	1

### B. Performance Evaluation

To demonstrate that our method works well in each task, we evaluate the average energy efficiency of 4 classical benchmarks from SPLASH-2, i.e. FFT, LU, CHOLESKY and RADIX. The energy efficiency of our method normalized to the traditional method's is shown in Table V. Our experiment shows the influence of selecting different inflection points on Fig. 3. When there're 3 demands regarded as hotspots, they don't cover the demands of some benchmarks. Thus, compared with the traditional method, there're 3 benchmarks that have no optimization. With the number of hotspots growing, the energy efficiency is increased.

TABLE V  
THE OPTIMIZED ENERGY EFFICIENCY.

Benchmarks	Hotspot Capacity			
	3	26	46	134
FFT	100.0%	99.3%	97.6%	82.0%
LU	100%	86.3%	86.1%	85.6%
CHOLESKY	88.4%	81.2%	80.7%	79.8%
RADIX	100%	97.7%	93.0%	82.7%

### C. Overhead Analysis

We use CACTI [14] in 32 nm technology to evaluate the overhead of our look-up table design from 4 aspects, i.e. area, access time, read energy and write energy. Since our experiment has 4 cores, 6 voltage levels and 15 current intervals, we set the tag bits as 84. The line size is 64 bytes. We show the overhead in Table VI. Note that the area of one core is 14.2997  $mm^2$ , the power of one core ranges from 0.2W to 6W, and the DVFS policy makes decisions every  $10^5$  ns. These data are obtained from GEM5 and McPAT. With the number of hotspots growing, the overhead is also increased. The designers should choose the highest capacity that they can accept the overhead to reach high enough optimization on the energy efficiency.

TABLE VI  
THE OVERHEAD OF OUR LOOK-UP TABLE DESIGN.

Overhead	Hotspot Capacity			
	3	26	46	134
Area ( $mm^2$ )	0.058	0.067	0.071	0.091
Access Time (ns)	0.37	0.39	0.40	0.45
Read Energy (nJ)	0.0298	0.0299	0.0301	0.0307
Write Energy (nJ)	0.0297	0.0310	0.324	0.0384

## IV. CONCLUSION

In this paper, we have proposed two techniques to optimize the energy efficiency of SCCs in MPSoCs, i.e. using look-up table to cache the predetermined optimized solutions and the hotspot recognition to handle various demands of load circuitries. The experiment explores the different hotspot capacities' effects on energy efficiency and overhead.

## REFERENCES

- [1] M. Ditty *et al.*, "NVIDIA's Tegra K1 system-on-chip," in *HCS*, 2014, pp. 1–26.
- [2] W. Wolf *et al.*, "Multiprocessor system-on-chip (MPSoC) technology," *TCAD*, vol. 27, no. 10, pp. 1701–1713, 2008.
- [3] T.-C. Chen *et al.*, "A biomedical multiprocessor SoC for closed-loop neuroprosthetic applications," in *ISSCC*, 2009, pp. 434–435.
- [4] W. L. Bircher and L. John, "Predictive power management for multi-core processors," in *ISCA*. Springer, 2010, pp. 243–255.
- [5] J. Jiang *et al.*, "A dual-symmetrical-output switched-capacitor converter with dynamic power cells and minimized cross regulation for application processors in 28nm cmos," 2017.
- [6] P. Zhou *et al.*, "Distributed on-chip switched-capacitor DC-DC converters supporting dvfs in multicore systems," *TVLSI*, vol. 22, no. 9, pp. 1954–1967, 2013.
- [7] L. Wang *et al.*, "Optimizing the energy efficiency of power supply in heterogeneous multicore chips with integrated switched-capacitor converters," in *DATE*, 2019, pp. 836–841.
- [8] V. Pallipadi and A. Starikovskiy, "The ondemand governor," in *Proceedings of the Linux Symposium*, vol. 2, no. 00216, 2006, pp. 215–230.
- [9] T. Guérout, T. Monteil, G. Da Costa, R. N. Calheiros, R. Buyya, and M. Alexandru, "Energy-aware simulation with DVFS," *Simulation Modelling Practice and Theory*, vol. 39, pp. 76–91, 2013.
- [10] N. Binkert *et al.*, "The GEM5 simulator," *ACM COMP AR*, vol. 39, no. 2, pp. 1–7, 2011.
- [11] S. C. Woo *et al.*, "The SPLASH-2 programs: Characterization and methodological considerations," *ACM COMP AR*, vol. 23, no. 2, pp. 24–36, 1995.
- [12] D. B. Witt, "Fully associate cache employing LRU groups for cache replacement and mechanism for selecting an LRU group," Dec. 12 2000, US Patent 6,161,167.
- [13] S. Li *et al.*, "McPAT: an integrated power, area, and timing modeling framework for multicore and manycore architectures," in *MICRO*, 2009, pp. 469–480.
- [14] R. Balasubramonian *et al.*, "CACTI 7: New tools for interconnect exploration in innovative off-chip memories," *TACO*, vol. 14, no. 2, pp. 1–25, 2017.