

Integrated Architectural/Physical Planning Approach for Minimization of Current Surge in High Performance Clock-gated Microprocessors

Yiran Chen
Purdue University
ECE School
465 Northwestern Ave.
W. Lafayette, IN 47907-2035
+1 765 494-1744
yc@ecn.purdue.edu

Kaushik Roy
Purdue University
ECE School
465 Northwestern Ave.
W. Lafayette, IN 47907-2035
+1 765 494-2361
kaushik@ecn.purdue.edu

Cheng-Kok Koh
Purdue University
ECE School
465 Northwestern Ave.
W. Lafayette, IN 47907-2035
+1 765 496-3683
chengkok@ecn.purdue.edu

Abstract

We propose an integrated architectural/physical planning approach to reduce the power supply noise due to current surge in high performance, general-purpose, clock-gated microprocessors. The proposed approach combines dynamic selection of functional units on-the-fly, dynamic issue width scaling and physical planning with soft module, to balance the current demand across layout. Experimental results show that the proposed approach could reduce the peak noise by 6.54% and consequently, the decoupling capacitance requirement by 21.8%. The degradation in IPC (Instruction Per Cycle) due to the selection logic and issue width scaling is only $1.86e-7$ (without increasing clock cycle period) in $0.18\mu\text{m}$ technology.

Categories and Subject Descriptors

B.7.1 [INTEGRATED CIRCUITS]: Types and Design Styles-*Microprocessors and microcomputers, VLSI*

General Terms: Design, Reliability

Keywords: Power supply noise, Inductive noise

1. Introduction

With higher degrees of integration and faster transistor switching speed, current-induced noise (IR drop and Ldi/dt noise) in the power supply network has become an important consideration in circuit design [1]. Recently, many proposed power management techniques such as clock gating, dynamic scaling of supply voltage (V_{dd}) [2] and transistor threshold voltage (V_t) [3] further aggravate the power supply noise problem as they introduce large amount of current surge. This paper focuses on the reduction of current surge due to clock gating technique.

Clock gating avoids dynamic power dissipation by disabling the clock to a circuit module whenever the circuit module is not being used. Reference [4] proposed a deterministic clock gating methodology (DCG) for the integer and floating point functional units (FUs) in superscalar microarchitecture. However, integer FUs have the highest current and power density [5] and they trigger large current surge in power supply network when they are

enabled/disabled simultaneously. The high peak noise generated by current surge severely limits the application of clock gating technology.

The power supply noise can be divided into three main voltage droops [6], which are due to the 1) high-frequency on-die current surge, 2) middle-frequency on-package current surge, and 3) low-frequency motherboard current surge, respectively. Usually the first and second voltage droops have significant amplitudes and both of them generate severe voltage drop in the power supply networks. In our paper, we focus on the reduction of the first power supply voltage droop and all subsequent references to the supply voltage droop refer to the on-die high frequency component.

Existing physical-design-level approaches on the current-induced noise reduction mainly include: (1) Assignment of the power pins [7]; (2) Sizing up the P/G networks [8]; and (3) Deployment of decoupling capacitance (Decap) [9]. However, these approaches introduce additional manufacturing cost or area and power overhead because they require more P/G pins, wider P/G networks or large leaky MOS decoupling capacitance. Recently, some physical design studies have begun to investigate other alternative methodologies to reduce the power supply noise without significant manufacturing cost, area and power overhead.

For example, in [10], authors notice that the distribution of current demands across the floorplan can affect the worst-case power supply noise. By optimizing the positions of high- and low-current-density circuit modules, the floorplanning algorithm proposed in [10] successfully balances the requirements of current in the floorplan and as a result, reduces the worst-case power supply noise generated by the current surge in the floorplan.

Some architectural methodologies also tried to solve current-induced power supply noise problem as well as to minimize the manufacturing cost or the area and the power overhead with acceptable performance penalty. By damping the pipeline of a high performance processor, the second power supply voltage droop described above is successfully reduced. However, the authors are unaware of any architectural methodology that addresses the on-die high frequency current surge.

In this paper, we propose an integrated architectural/physical planning approach to minimize current surge due to integer FUs in DCG microprocessors. We introduce a function unit selection scheme to balance the current demands across the integer FUs dynamically. As a result, the peak noise in power supply network is reduced. Moreover, we propose two complementary methodologies to further enhance the effectiveness of the proposed dynamic FU selection scheme with minimal performance penalty. They are

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

ISLPED '03, August 25–27, 2003, Seoul, Korea.

Copyright 2003 ACM 1-58113-682-X/03/0008...\$5.00.

dynamic issue width scaling and physical planning with soft module.

Our experimental results show that the proposed approach can reduce the peak noise by 6.54% and consequently, the decoupling capacitance requirement by 21.8% for 0.18 μ m technology. The worst-case performance loss (in terms of Instructions Per Cycle (IPC) degradation) is only 1.86e-7 with no increase in clock cycle

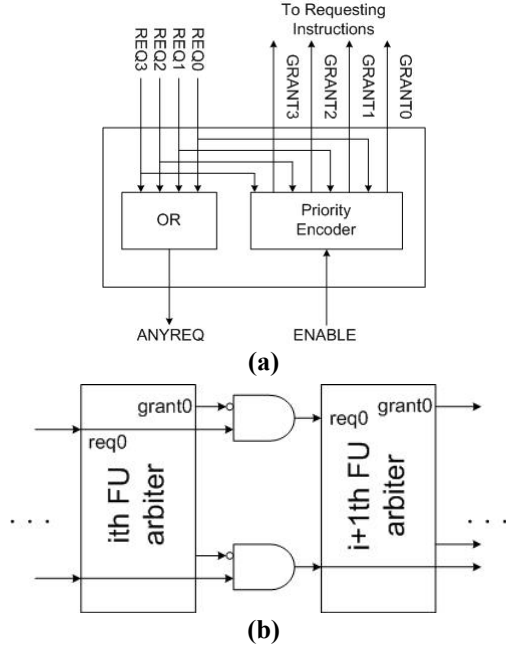


Fig. 2 Schematic of stacked selection logic
(a) FU arbiter cell (b) Stacked FU arbiter cells

period.

2. Current surge in DCG microprocessor

As in [4], we assume that a clock-gated FU does not draw any current (except leakage current) from power pins when it is not used in the current cycle, i.e., when it is “quiet”. Also, when an FU is used in the current cycle, i.e., it is “active” and it draws the peak current.

Fig. 1 shows two popular floorplans for a 6-way, non-clustered pipeline [12, 13]. Each configuration includes two general ALUs (GUs), four simple integer ALUs (SUs), and two 3-stage, pipelined integer multipliers (MULTs). A GU consists of an adder, a shifter, and a logic unit whereas an SU has only an adder and a logic unit. To reduce the length of bypass wire, MULTs are usually placed at both ends. The main difference between the two floorplans is the

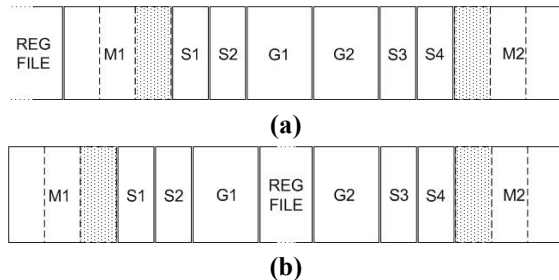


Fig. 1 Floorplans of integer FUs

position of the register file (REG FILE). We use G, S, and M to denote GU, SU, and MULT, respectively, in Fig. 1.

We refer to the numbers and types of FUs that are used in the current cycle as the “*issue pattern*”. “2G3S1M”, for example, denotes the issue pattern that 2 GUs, 3 SUs and 1 MULT are needed for instructions issued in the current cycle. In the present-day, wide issue, high performance pipeline, the issue pattern is determined by the selection logic in the wakeup/selection (issue) stage, which applies a certain selection policy to select a subset of the instructions that are ready for execution and the corresponding FUs to execute them.

A schematic of the selection logic is shown in Fig. 2. An FU arbiter (Fig. 2(a)) asserts the ANYREQ signal to enable the corresponding FU if any of its input request signals (REQ) is high. The FU arbiter also applies some selection policy on the input request signals to generate the grant signals (GRANT) that are sent back to corresponding entries in the issue window. For single-cycle latency or pipelined multi-stage integer FUs, the ENABLE signal is always asserted. When there are multiple functional units of the same type, the selection logic is made up of a number of stacked FU arbiters as shown in Fig. 2(b). When a request signal (e.g., req0) is granted by the *i*th FU arbiter, it will be masked by the corresponding GRANT signal (e.g., grant0) and not reach the (*i*+1)th FU arbiter. As a result, every FU arbiter grants one instruction, and the *i*th FU is not selected unless the first *i*-1 FUs are not available.

However, stacked selection logic design is not optimal from the point of view of minimizing current surge and peak noise in power supply network. In this paper, we refer to the combination of the physical FUs used to execute the issue pattern as the “*usage pattern*” of FUs. In Fig. 1(a), for example, the usage pattern “M1+S1+S2+G1+G2+S3” will be selected by the stacked selection logic for issue pattern 2G3S1M, if we assume that the selection priorities of FUs of the same type are ordered from left to right. As FUs M1, S1, S2, G1, G2 and S3 are clustered together, the current demands in layout are very unbalanced. Here we assume that each of M1 and M2 has 3 pipelined stages (without loss of generality) which can be clock-gated separately.

3. Architectural/physical planning approach

3.1 Dynamic FU selection

We note that the usage pattern selected above is not the only choice for the issue pattern 2G3S1M. There are some alternative usage patterns that induce lower peak noise level due to better balanced current demands in the floorplan. All alternative usage patterns for the issue pattern 2G3S1M for the floorplan in Fig. 1(a) with corresponding peak noise are shown in Table 1. These results are simulated by HSPICE for 0.18 μ m technology (see Section 5). For the analysis of the worst-case peak noise, we assume that the last two stages of the two MULTs are used.

Obviously, if we can choose the optimal usage pattern for every possible issue pattern cycle by cycle, the induced peak noise can be minimized. We refer to this method as the dynamic FU selection scheme. Essentially, *this scheme minimizes the peak noise among all usage patterns for a given issue pattern.*

In general, the highest peak noise occurs when the entire issue width is used and all stages of multi-stage FUs are active (except for the input stages). For simplicity, we omit the behavior of the peripheral circuits of FUs although they also contribute to the worst-case peak noise. To avoid the possible inaccuracy introduced by this simplicity, we only assume highest current densities of FUs and peripheral circuit in the peak noise simulation. We point out this assumption

does not cause overestimation of noise because only the worst-case power supply noise determines the reliability of the system and the requirement of decoupling capacitance.

Table 1 Alternative usage patterns and peak noises

Usage pattern	Peak noise (V)
M1+S2+G1+G2+S3+S4	0.2961
M1+S1+G1+G2+S3+S4	0.2942
M1+S1+S2+G1+G2+S4	0.3009
M1+S1+S2+G1+G2+S3	0.3023
S2+G1+G2+S3+S4+M2	0.3021
S1+G1+G2+S3+S4+M2	0.3003
S1+S2+G1+G2+S4+M2	0.2944
S1+S2+G1+G2+S3+M2	0.2962

We note that dynamic FU selection scheme does not introduce any IPC-based (instruction per cycle-based) performance loss except for the potential increase in the clock period due to the complexity of new logic to perform dynamic FU selection. In Section 4, however, we shall show that for the 0.18 μ m technology and beyond, this method does not extend the clock period.

3.2 Dynamic issue width scaling

There are issue patterns that have only one usage pattern each; they cannot be optimized by dynamic FU selection. If these issue patterns generate the worst-case peak noise among all issue patterns, *the reduction of peak noise cannot be achieved*. This situation usually occurs when all of the FUs involved in this issue pattern are clustered together. Dynamic FU selection scheme is ineffective in these cases. Hence, we propose a dynamic issue width scaling methodology: The selection logic holds back some instructions to prevent the worst case issue pattern from occurring.

Dynamic issue width scaling possibly incurs some performance penalty because of the degradation of pipeline issue width utilization. But our experimental results in Section 5 show that the worst case to which we need to apply dynamic issue width scaling rarely happens, i.e., the impact on performance is negligible. Note however, similar as the discussion in section 3.1 that, the decoupling capacitance requirement is determined by the worst-case peak noise, regardless of its frequency of occurrence.

3.3 Physical planning with soft module design

Another constraint for dynamic FU selection methodology is the existence of large FU with high current demand. For example, a multi-stage pipelined multiplier is comparably large. To reduce the data transferring latency between sequential stages, all stages of it are placed close to each other. As the worst case usually happens when multi-stage FUs were fully loaded, the effectiveness of the dynamic FU selection scheme degrades.

One solution is to subdivide FU into several non-contiguous blocks, i.e., soft modules. A stage in a multi-stage block, for example, is a natural soft module. This methodology increases the flexibility of physical planning in two ways: First, it reduces the sizes of high-current-demand modules. Second, “quiet” modules can be scattered across the floorplan to balance the distribution of current demands.

Soft module design introduces routing congestion and timing performance problems. For example, it increases the data transfer latency from one stage to the subsequent stage. If the soft module lies on a critical path, the clock period has to increase. Fortunately, the data transfer between different stages in multi-stage FUs is not in the critical path of the pipeline [12, 13, 14]. Soft module design is

well-studied research area. For more details, readers can refer to the corresponding references.

4. Implementation issues

4.1 Dynamic FU selection logic

To implement the dynamic FU selection scheme, we can make a small modification to any existing selection logic by adding a small *look-up table*. The tag and the corresponding entries of the look-up table are determined as follows: The power supply noise level that can be tolerated by the design is denoted as “TNL” (tolerable noise level). If the usage pattern selected by the selection logic for a particular issue pattern induces a peak noise that exceeds TNL, we store that usage pattern as a tag in look-up table. The optimal usage pattern (see Section 3.1) for that issue pattern is stored as the corresponding entry in the lookup table. The new selection logic works as follows: If the usage pattern UP generated by the selection logic can match a tag in the look-up table, the corresponding optimal usage pattern UP_O will be selected as the final usage pattern UP_V (see Fig. 3). Otherwise, the usage pattern UP is selected since it does not induce a noise level that exceeds TNL.

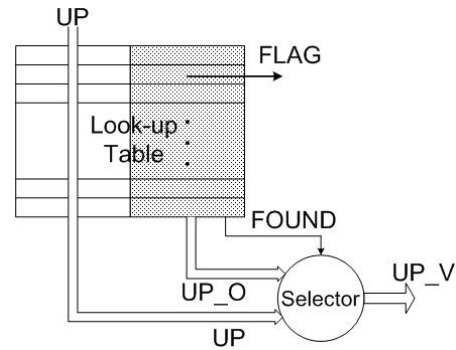


Fig. 3 Scheme of look-up table

Since we do not store all possible usage patterns generated by the original selection logic in the look-up table, the size of the look-up table is small. Our experiments with a 6-way baseline microprocessor show that a 20-entry table is sufficient (see Section 5.2).

Obviously, the existence of the look-up table prolongs the latency of wakeup/selection stage. If the wakeup/selection stage is the crucial stage in the pipeline, performance loss may be introduced due to the increase in cycle period. We performed an analysis of the latencies of a 20-entry look-up table and wakeup/selection stage for a 6-way microprocessor at 0.18 μ m technology based on the model and the configuration in [12]. The analysis (see Table 2) shows that the added complexity of the selection logic will not incur performance loss for the reason that execution/bypass stage is the crucial stage in the pipeline. This is the same conclusion drawn in [12, 13, 14].

Table 2 Latency increasing of selection logic

Technology	0.18 μ m
Look-up table latency (ps)	32.9
Orig. wakeup/selection stage latency (ps)	815.9
New wakeup/selection stage latency (ps)	848.8
execution/bypass stage latency (ps)	850.7

4.2 Dynamic issue width scaling

To implement the dynamic issue width scaling scheme, we augment the entries of the look-up table with additional signal “FLAG”.

As in Fig. 4, when usage pattern to which we need to apply dynamic issue width scaling is found, a FLAG signal is asserted (see Fig. 3). If the history status of multi-stage FUs indicates that the peak noise induced by the optimal usage pattern exceeds the TNL, a DISABLE signal is generated to deny a GRANT signal to the offending instruction.

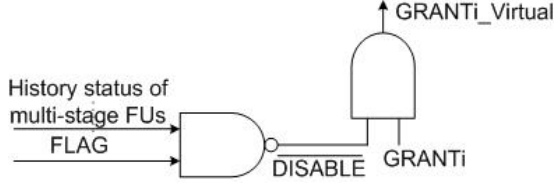


Fig. 4 Dynamic issue width scaling logic

4.3 Soft module design

The natural implementation for soft module design is to divide multi-stage FUs into several separate stages. Usually, only the first stage should be designed as soft module because other stages are always considered to be used when estimating the worst-case peak noise. Hence, only the first stage could be quiet and helps to balance the distribution of current demand when dynamic FU selection mechanism applied.

We use {RMMSSGGSSMMM} and {MMMSSGRGSSMMM} to denote the two floorplans in Fig. 1(a) and (b), where R, M, S, and G denote REG FILE, MULT, SU and GU, respectively. Here we distinguish the two 3-stage MULTs by “MMM” and “MMM”. The optimized floorplans we obtain for Fig. 1(a) and (b) are {RMMSSMGGMSSMM} and {MMSMSGRGMSSMM}, respectively. We note that an additional wire length of 306 μ m is introduced between the 1st and 2nd stages of MULT.

5. Experiments and Results

Architectural level and physical level experiments are implemented with various combinations of our three methodologies:

- DFS – dynamic FU selection;
- DIWS – dynamic issue width scaling;
- SMD –physical planning with soft module.

The power supply V_{dd} is 1.8V. The TNL is set to 10% of V_{dd} , which is 0.18V in our experiments.

5.1 Peak noise evaluation

HSPICE is used for the evaluation of peak noise at 0.18 μ m technology.

For the 6-way microprocessor in our experiments, two floorplans of integer FU in Fig. 1 are considered. Table 3 summaries the dimensions of FUs [12], where λ is half the feature size. In the floorplan, the row heights of MUXs (bypass logic) between functional units are ignored. We estimate power supply noise by using the same simulation methodology accepted in [10]. The pitch for the metal lines in the power supply is 240 μ m and the pitch for VDD pins is 720 μ m. The relative technology parameters are shown in Table 4. The current density J_s is set to 0.28 μ A/ μ m² at 0.18 μ m technology, which is estimated based on ITRS’99 Roadmap [15]. We assume that every power grid point k supplies current to an area delimited by the four bisectors defined by the point k and its four

adjacent neighbors. The peak switching current drawn at point k is calculated as $I^{(k)} = \sum_j factor[j] \cdot J_s \cdot A_{jk}$, where A_{jk} is the

delimited area between power grid point k and circuit module j . $factor[j]$ depends on the module j ’s behavior. For integer FUs, $factor[j]$ is set to 0 or 2 in clock-gated or un-gated cycle, respectively; for other circuit modules, such as REG FILE, the factor is set to 1 because of their low current demands. The overall switching current waveform at power grid point k is approximated with a triangular waveform with peak value $I^{(k)}$ and the duration of switching current waveform (τ) is assumed to be half the clock period (1ns). Based on the discussion in [1], we believe that triangular current waveform gives a more accurate description of the switching current. We evenly distribute the decoupling capacitance at all nodes of the power mesh, as assumed in [16]. We do note that the allocation of Decap also affects its effectiveness on the noise reduction. However, we have not considered Decap allocation – it is beyond the scope of this paper.

In our simulation, at first the current demand distribution across floorplan for every possible usage pattern is generated based on the usage information of every FU. Then the generated current waveforms at every node are put into HSPICE to simulate the power supply noise. In the Decap requirement estimation step, the power supply noise limitation is set as 10% of power supply voltage (1.8V). The value of Decap needed for 0.18V noise-limit is considered as the Decap requirement (with even Decap distribution as described above).

Table 3 Functional unit dimensions

FU	Width	Height	Description
GU	4732 λ	3200 λ	adder, shifter and logic unit
SU	4732 λ	1700 λ	adder and logic unit
MULT	4732 λ	5100 λ	3-stage multiplier
REG	5900 λ	21120 λ	96-entry

Table 4 Technology parameters

Para.	Description	Value
r	Wire resistance per unit length (Ω/μ m)	0.0241
l	Wire inductance per unit length (pH/μ m)	0.8
c	Wire capacitance per unit length (pF/μ m)	20
L_p	Package inductance per VDD pin (nH)	0.2
R_p	Package resistance per VDD pin (Ω)	0.5

Tables 5 and Table 6 show the experimental results for the floorplans in Fig. 1(a) and (b), respectively. The column ORIG in each table represents the results without applying any optimization. The entries in the second rows “Peak noise” of Tables 5 and 6 are the peak noise without decoupling capacitance in the power mesh. The amounts of Decap requirements to keep the peak noise below TNL for each case are shown in the rows “Decap”. For the floorplan in Fig. 1(a), the original peak noise of worst cases is 0.3023V, which is induced by issue pattern 2G3S1M. For the same issue pattern, the peak noise of the floorplan in Fig. 1(b) is 0.2916V. Their decoupling capacitance requirements for the 0.18V power-supply noise limitation are 2.604nF and 2.311nF, respectively. The Decap reductions are shown in the rows “Decap reduction”, which are obtained with respect to the original Decap requirements. The rows “worst case issue pattern” include the issue patterns that generate the worst-case peak noise and determine the amount of Decap requirement.

Table 5 Experimental results for configuration in Fig. 1 (a)

Methodologies	ORIG	DFS	DFS+DIWS	DFS+SMD	DFS+SMD+DIWS
Peak noise (V)	0.3023	0.2994	0.2942	0.2848	0.2828
Peak noise reduction	/	0.96%	2.68%	5.79%	6.45%
Decap (nF)	2.604	2.487	2.363	2.107	2.037
Decap reduction	/	4.5%	9.3%	19.1%	21.8%
Worst case issue pattern	2G3S1M	2G4S0M	2G3S1M	2G2S2M	2G3S1M
Largest IPC-based perf. loss	/	0	1.86e-7	0	6.38e-8

Table 6 Experimental results for configuration in Fig. 1 (b)

Methodologies	ORIG	DFS	DFS+DIWS	DFS+SMD	DFS+SMD+DIWS
Peak noise (V)	0.2916	0.2811	0.2784	0.2805	0.2784
Peak noise reduction	/	3.60%	4.53%	3.81%	4.53%
Decap (nF)	2.311	2.092	2.012	2.073	2.012
Decap reduction	/	9.5%	13.6%	10.3%	13.6%
Worst case Issue pattern	2G3S1M	2G4S0M	1G3S2M	2G2S2M	1G3S2M
Largest IPC-based perf. loss	/	0	1.86e-7	0	1.86e-7

• **DFS**

The DFS columns in the tables show the results after applying DFS. The peak noise levels are reduced for both floorplans. DFS is less effective for the floorplan in Fig. 1(a) as a consequence of the unavailability of alternative usage patterns for the new worst-case issue pattern 2G4S0M. However, this problem is overcome successfully after we apply SMD (see the result analysis in SMD part).

• **DIWS**

The more issue patterns we apply DIWS to, the more Decap requirement reduction we can get, however, the more performance penalty we should pay. The effectiveness of DIWS depends on the difference of peak noises generated by the original and new worst cases before and after DIWS applied. In our experiments, we apply DIWS to one issue pattern for the floorplan in Fig. 1(a) and three issue patterns in the floorplan in Fig. 1(b) since they generate similar distribution of current demands in layout. Our results show that the DIWS is very effective for the floorplan in Fig. 1(a). The analysis of IPC-based performance loss of DIWS is given in Section 5.2.

• **SMD**

The optimized floorplans via SMD are illustrated in Section 4.3. As the results in the DFS+SMD and DFS+SMD+DIWS rows in Tables 5 and 6 show, the reductions in peak noise and Decap are significant. Also, by comparing the corresponding results (DFS and DFS+SMD, DFS+DIWS and DFS+SMD+DIWS), we know that SMD successfully enhance effectiveness of existing methodologies by breaking down MULTs.

5.2 IPC-based performance evaluation

The modified SimpleScalar Tool [17] is used for architectural level simulation for DIWS. The configuration of the baseline pipeline is shown in Table 7. It is a 6-way microprocessor with 4 SUs, 2 GUs, 2 MULTs and a 96-entry REG FILE. We used pre-compiled Alpha Spec2000 binaries [18] to analyze the IPC-based performance. The benchmarks are compiled with SPEC *peak* setting. For each of the benchmarks, we used *ref* inputs, skipped the first 2 billion instructions, and simulated 500 million instructions (as in [4]).

Table 7 Baseline processor configuration

Processor	6-way issue, 96-entry RUU, 64-entry load/store queue, 4 int. SUs, 2 int. GUs, 2 int. MULTs, 3 FP ALUs, 3 FP MULTs, 96-entry REG.
Branch prediction	2-level, 8192-entry in 1st level, 8192-entry in 2nd level, 2B history; 32-entry RAS, 8192-entry 4-way BTB, 8 cycle mispredict penalty
Caches	64KB 2-way 2-cycle I/D L1, 2MB 8-way 12-cycle L2, both LRU
Main memory	Infinite capacity, 100 cycle latency; Split transaction, 32-byte wide bus

The worst case IPC-based performance penalties due to the dynamic issue width scaling scheme for the floorplans in Fig. 1(a) and (b) are all 1.86e-7, over 10 integer benchmarks and 12 float point benchmarks. The results indicate that we rarely have to apply dynamic issue width scaling and as a result, processor seldom uses all its resource for most applications. However, the scheme is absolutely necessary for keeping the peak noise below the TNL.

6. Conclusion

In the paper, an integrated architectural/physical planning approach is proposed to reduce (1) the power supply noise due to current surge and (2) the decoupling capacitance requirement in high performance, general-purpose, clock-gated microprocessors. The proposed approach combines three techniques including dynamic selection of functional units in run-time, dynamic issue width scaling, and physical planning with soft module to balance the current demands across floorplan. Experimental results show that our approach reduces the peak noise by 6.54% and the decoupling capacitance requirement by 21.8% at 0.18μm technology. Our results also show that worst-case overall performance loss in IPC is only 1.86e-7 without increasing clock cycle period. Although our simulation and results in this paper are only for the power supply noise reduction of integer FUs, we believe our methodology can be extended to other pipeline stages and relevant hardware.

References

[1] H. H. Chen and J. S. Neely. Interconnect and circuit modeling techniques for full-Chip power supply noise analysis. *IEEE*

- Trans. Component, Packing and Manufacturing Tech.*, Part B, vol. 21, pages 209-215, Aug. 1998.
- [2] J. Pouwelse, and K. Langendoen, and H. Sips. Dynamic voltage scaling on a low-power microprocessor. *7th International Conference on Mobile Computing and Networking (Mobicom)*, pages 251-259, Jul. 2001.
- [3] C. H. Kim and K. Roy. Dynamic Vt SRAM: a leakage tolerant cache memory for low voltage microprocessors. *In proceedings of the 2002 International Symposium on Low Power Electronics and Design (ISLPED)*, pages 251-254.
- [4] H. Li, S. Bhunia, Y. Chen, T. N. Vijaykumar, and K. Roy. Deterministic clock gating for microprocessor power reduction. *In proceedings of the 9th International Symposium on High Performance Computer Architecture (HPCA)*, pages 113-124, Feb. 2003.
- [5] K. Wilcox, and S. Manne. Alpha processors: a history of power issues and a look to the future. *Cool Chips Tutorial, in Conjunction with the 32nd Annual International Symposium on Microarchitecture*, Nov. 1999.
- [6] Lseley Polka, etc., Package-Level Interconnect Design for Optimum Electrical Performance. Intel Technology Journal Q3, 2000.
- [7] U.A Shrivastava and B.L. Bui. Inductance calculation and optimal pin assignment for the design of pin-grid-array and chip carrier packages. *In IEEE Transactions on Components, Hybrids, and Manufacturing Technology*, Part A, B, C , Volume: 13 Issue: 1 , pages: 147 -153, Mar. 1990.
- [8] T.-Y. Wang and C.C.-P Chen. Optimization of the power/ground network wire-sizing and spacing based on sequential network simplex algorithm. *In Proceedings of International Symposium on Quality Electronic Design*, pages: 157 -162, 2002.
- [9] R. Downing, P. Gebler and G. Katopis, Decoupling capacitor effects on switching noise. *In Electrical Performance of Electronic Packaging*, pages 148 -150, 22-24 Apr. 1992.
- [10] S. Zhao, K. Roy and C.-K. Koh, "Decoupling Capacitance Allocation and Its Application to *Power-Supply Noise-Aware Floorplanning*", *IEEE Trans. Computer-Aided Design*, vol. 81, pp.81-92, Jan. 2002.
- [11] Michael Powell and T. N. Vijaykumar, Pipeline Damping: A microarchitecture technique to reduce inductive noise in supply voltage. *In Proceedings of the 30th Annual International Symposium on Computer Architecture (ISCA)*, Jun. 2003.
- [12] S. Palacharla, N. P. Jouppi and J. E. Smith. Quantifying the complexity of superscalar processors. *Technical report CS-TR-96-1038*, Dept. of CS., Univ. of Wisconsin, 1996.
- [13] E. S. Fetzer, M. Gibson, A. Klein, N. Calick, C. Zhu, E. Busta, and B. Mohammad. A fully bypassed six-issue integer datapath and register file on the itanium-2 microprocessor. *IEEE Journal of Solid-State Circuits*, 37(11), pages 1433-1440, Nov. 2002.
- [14] L. T. Clark, E. J. Hoffman, J. Miller, M. Biyani, L. Liao, S. Strazdus, M. Morrow, K. E. Velarde, and M. A. Yarch. An embedded 32-b microprocessor core for low-power and high-performance applications. *IEEE Journal of Solid-State Circuits*, 36(11), pages 1599-1608, Nov. 2001.
- [15] International Technology Roadmap for Semiconductor, 1999.
- [16] M. D. Pant, P. Pant, and D. S. Wills. On-chip decoupling capacitor optimization using architectural level current signature prediction. *13th Annual IEEE International ASIC/SOC Conference*, pages 288-292, 2000.
- [17] D. Burger and T. M. Austin. The simplescalar tool set, version 2.0. TR1342, University of Wisconsin, Jun. 1997.
- [18] D. Weaver, "Pre-compiled little-endian Alpha ISA SPEC2000 binaries", <http://www.eecs.umich.edu/~chriswea/benchmarks/spec2000.html>