UC Santa Cruz
UC Santa Cruz Previously Published Works

Title
A Decap Placement Methodology for Reducing Joule Heating and Temperature in PSN Interconnect

Permalink
https://escholarship.org/uc/item/56t894b2

Authors
Guthaus, Matthew
Logan, Sheldon

Publication Date
2013-08-01

Peer reviewed

eScholarship.org Powered by the California Digital Library
University of California

A Decap Placement Methodology for Reducing Joule Heating and Temperature in PSN Interconnect

Sheldon Logan, Matthew R. Guthaus
Department of CE, University of California Santa Cruz, Santa Cruz, CA 95064
{slogan,mrg}@soe.ucsc.edu

Abstract— **Power Supply Networks (PSN) are susceptible to electromigration failure and increased resistance due to high on-chip temperatures and large power supply currents. Joule heating, which leads to increased localized interconnect temperatures and higher resistivity in the PSN interconnect, exacerbates this reliability problem and is only expected to worsen in future technologies. The best method of reducing interconnect Joule heating is by reducing the RMS current within the interconnect. Consequently, we propose the first gradient-based decoupling capacitance placement method to reduce the magnitude of the current spikes in the interconnect. Our experiments show that our propose approach can reduce interconnect temperatures by 12.5 K which results in a 4.7% decrease in resistivity and an increase of 66.3% in electromigration lifetime.**

I. Introduction

Power Supply Network (PSN) interconnect reliability has become a growing concern among Integrated Circuit (IC) designers due to the increased currents and current densities required in most designs. These lead to large temperature increases in the interconnect [3] caused by resistive Joule heating which lead to increases in interconnect resistivity and increases in the rate of electromigration in the interconnect. The increased resistance only exacerbates the Joule heating problem and can cause IR drop and voltage droop which in turns leads to timing and signal integrity issues [14].

The increases in current and current density are going to worsen as designs move to smaller technologies. Decreased wire cross section areas, lower supply voltages and low-K dielectrics between metal layers all increase the temperature impact of Joule heating [4]–[6].

Researchers have attempted to solve the temperature problem for PSN interconnect in various ways. Wang et al. proposed a wire sizing algorithm considering the thermal impact of electromigration and Joule heating [15]. However, since there are limited wire resources in modern designs, wire sizing can only alleviate the thermal issues in the PSN network by a small amount. Lele et al. proposed a method to optimally size global interconnect wires and signal repeaters [7], but this method is not applicable to PSN interconnects which do not use repeaters. Yokogawa et al. demonstrated that stacking vias between the metal wires and the substrate can help mitigate temperature increases in PSN interconnect due to increased thermal conductivity between the wires and substrate [16].

Another possible method of solving the current and current density problem is to decrease the RMS current in the wires and the distance that the current has to travel through the wires. Most power is supplied from the package bumps/pins, but, unfortunately, this power must go through the various metal interconnect layers before reaching the transistors on the substrate. This can lead to a significant amount of energy loss due to Joule heating through the PSN wires. However, power is also temporarily supplied to the IC using decoupling capacitances (decaps) which are attached to the lowest metal level that is much closer to the switching devices. To take advantage of this, we propose the first method of placing decaps in areas that require significant power to reduced the Joule heating in the interconnect by reducing the distance in wires that the power supply current has to travel and also reducing the current spikes in the wires. Both of these lead to smaller RMS current values in PSN interconnect.

All previous decap algorithms proposed in literature have focused on minimizing voltage droop [8], [9], [11], [13]. Our work is the first to present a method of redistributing decaps to reduce the Joule heating power *while still considering traditional voltage droop*. Consequently, the temperatures of PSN interconnects are decreased and the reliability and signal integrity are increased. We use a gradient-based algorithm to determine the best method of redistributing decaps to reduce Joule heating power and then extend our algorithm to consider reducing interconnect temperatures directly. Specifically, this is the first work

- to analyze the impact of Joule heating on PSN interconnect reliability,
- to use decap to reduce Joule heating in the PSN,
- to propose a gradient based algorithm to guide decap redistribution in order to reduce Joule heating and wire temperatures in PSN interconnects.

Our work proceeds as follows: Section II contains background information on PSNs, Joule heating and electromigration. Section III introduces the method of using decap placement to reduce Joule heating. Section IV provides the experimental setup and then the results obtained from our method are presented in Section V. Finally, Section VI concludes our results.

II. Power Supply Networks

A PSN consists of power pins from the package, wires (interconnect) and decoupling capacitors (decaps). The power pins and the decaps supply current to transistors in the circuit. The power pins are modeled as inductors in series with resistors attached to an ideal, off-chip voltage source. The interconnect distributes the current throughout the circuit from the power pins/decaps and is modeled as a network of resistors. The transistors are modeled as individual, distributed current sources with a small amount of diffusion capacitance. The decaps act as a local energy supply and assist in reducing the dynamic voltage droop caused by the sudden current draw of the localized transistor switching.

A. Electrical Simulation

The voltages for a PSN are calculated using Modified Nodal Analysis (MNA) according to

$$G \cdot v(t) + C \cdot \dot{v}(t) = i(t) \quad (1)$$

where G is the conductance matrix, C is the admittance matrix (inductance and capacitance elements), $v(t)$ represents the time varying nodal voltages and $i(t)$ represents the vector of current sources corresponding to the transistors in the design.

B. Joule Heating

Not all power supplied to an IC is converted to useful switching energy. Energy is lost, in the form of Joule heating while distributing current throughout the circuit due to interconnect resistance. The power loss to Joule heating at a given time, $J(t)$, in each wire/interconnect segment is

$$J(t) = i^2(t) R \tag{2}$$

where R represents the resistance of the interconnect segment and $i(t)$ is the current through the segment.

The temperature increase, ΔT_{wire}, in a wire segment due to this power loss is proportional to the root mean square (RMS) value of the Joule heating power [3] and is computed using

$$\Delta T_{\text{wire}} = \frac{RR_\theta}{D} \int_0^D i^2(t)\, \mathrm{d}t. = i_{\text{RMS}}^2(t) RR_\theta \tag{3}$$

where i_{RMS} is the RMS value of the current, R_θ is the thermal resistance of a wire to the substrate, R is the resistance of the interconnect, and D is the duration over which the Joule heating is being analyzed.

The thermal resistance, R_θ, is measured with respect to the substrate since the majority of heat produced in an IC is removed from the heat sink attached to the back-side of the substrate. This thermal resistance can be estimated [3] as

$$R_\theta = \frac{t_{\text{ins}}}{K_{\text{eff}} L W_{\text{eff}}} \tag{4}$$

where t_{ins} is the thickness of the insulation between the metal wire and the substrate, K_{eff} is the effective thermal conductivity of the thermal insulation, L is the length of the wire and W_{eff} is the effective width of the wire.

The final temperature of each interconnect element depends on both the global temperature due to dynamic switching and the corresponding heat from the substrate along with the local Joule heating according to

$$T_{\text{wire}} = T_{\text{sub}} + \Delta T_{\text{wire}} \tag{5}$$

where T_{sub} is the temperature due to the substrate directly below the wire and ΔT_{wire} was defined in Equation 3.

Once the final wire temperatures are computed, the increase in wire resistance (ΔR) caused by Joule heating is calculated from the definition of resistivity as

$$\Delta R = \frac{1 + \alpha (T_{new} - T_{ref})}{1 + \alpha (T_{old} - T_{ref})} \tag{6}$$

where T_{old} to T_{new} are the initial and final wire temperature respectively, α is the thermal coefficient of resistivity, and T_{ref} is the reference temperature for the thermal resistivity coefficient.

C. Electromigration of PSN Interconnect

The Mean Time to Failure (MTTF) of a metal wire due to electromigration is calculated using Black's Equation,

$$\text{MTTF} = A \frac{1}{j_{\text{avg}}^n} \exp\left(\frac{Q}{kT_{\text{wire}}}\right), \tag{7}$$

where A is a constant based on the cross-sectional area of the wire, j_{avg} is the average current density ($\frac{A}{cm^2}$), Q is the activation energy, n is a fitted model parameter, k is the Boltzmann's constant, and T_{wire} is the wire temperature from Equation 5. The electromigration failure rate in wires is exponentially dependent on temperature hence it is critical to reduce the Joule heating in wires. If the temperature of wire changes from temperature T_{old} to T_{new} the MTTF failure rate decrease can be calculated from Equation 7 as

$$\Delta\text{MTTF} = \exp\left(\frac{Q}{k}\left(\frac{1}{T_{old}} - \frac{1}{T_{new}}\right)\right). \tag{8}$$

III. Decap Redistribution to Reduce Interconnect Temperatures

Decap is a viable mechanism for reducing Joule heating in wires, and consequently interconnect temperatures, because it decreases the magnitude of the current spikes and consequently reduces the Joule heating RMS value. Decap is usually added to a design to reduce transient voltage droop, however, this decap placement is usually quite flexible with multiple placements meeting the voltage droop requirement. Therefore, we propose to redistribute decap after initial placement for voltage droop while minimizing wire temperatures. Additional decap is added in areas that have high wire temperatures while decap is removed from areas that do not have significant high interconnect temperatures and are well within the voltage droop bounds in order to preserve PSN integrity.

Our proposed method of decap redistribution uses a gradient descent, non-linear optimization. The design is spatially partitioned so that the sensitivity to reduce interconnect temperatures of each partition with respect to the decap value can be measured. This enables decap to be shifted from less sensitive partitions to the more sensitive partitions. A detailed implementation of our algorithm is presented in Algorithm 1.

The first stage of the algorithm on Lines 1-3 calculates the total decap redistribution budget, C_{budget}. The design is partitioned into various regions at the block level or potentially a finer granularity. Next, each partition of the PSN is simulated with a small, fixed amount of decap removed from that partition to determine if removing decap causes a voltage droop violation. If so, that partition is flagged so that decap can only be added to it and not removed. If removing decap does not cause a violating voltage droop in the partition, the decap is permanently removed and added as surplus to the overall decap redistribution budget, C_{budget}. The amount of decap removed during each simulation, C_{rem}, is a variable within the algorithm. Large values for C_{rem} limit the number of partitions that contribute to C_{budget} since removing a large amount of decap will likely cause excessive voltage droop. On the other hand, small values of C_{rem} limit the effectives of the algorithm due to the small amounts of decap added to the redistribution budget. Finally, C_{budget} is then uniformly distributed to the redistribution decap budget (δC_i) for each partition as an initial redistribution which is subsequently improved on (Lines 4-11). The initial decap in each partition's δC_i is consequently $\frac{C_{\text{budget}}}{N}$ where N is the number of partitions.

The refinement stage on Lines 4-11 begins by calculating the sensitivity of the maximum wire temperature in each partition to the decap in that partition, $\frac{dT_i}{dC_i}$, on Line 5. This determines which partitions will most (or least) benefit from additional decap. The sensitivities are measured by simulating the entire PSN with the present decap allocation and calculating the maximum wire temperature, T_i^{part1}, within each partition i. The PSN is then re-simulated with each partition containing a small amount of added decap, δ_{add}, to calculate a forward finite difference. The maximum wire temperature for each partition with the added decap is calculated as T_i^{part2}. The average decap sensitive for a partition can thus be calculated as:

$$\frac{dT_i}{dC_i} = \frac{T_i^{\text{part2}} - T_i^{\text{part1}}}{\delta_{add} N_{wire}} \tag{9}$$

Algorithm 1 Decap Redistribution to Minimize Joule Heating

Input: Decap placement satisfying voltage droop requirements.
Output: Decap placement minimizing total Joule heating while still satisfying voltage droop requirements.

1: Partition design into N partitions
2: Determine total redistribution budget (C_{budget})
3: Uniformly allocate C_{budget} to each partition's redistribution budget (δC_i)
4: **repeat**
5: Calculate sensitives ($\frac{dT_i}{dC_i}$) for each partition
6: Calculate mean of sensitivities
7: Find low sensitivity partitions (sensitivity value below mean)
8: Find high sensitivity partitions (sensitivity value above mean)
9: Redistribute decap from the low sensitivity partitions to the high sensitivity partitions
10: Calculate the max wire temperature across all partitions $T_{\max}$
11: **until** $T_{\max} \leq \epsilon$

where N_{wire} refers to the number of wires within the partition and C_i refers to the decap in partition i.

The final stage of the algorithm on Lines 6-11 determines how to incrementally redistributed the decap, C_{budget}, across all the partitions. First, the mean of the partition sensitivities is computed on Line 6. All partitions with sensitivities above the mean are flagged as partitions to receive more decap and those partitions below the mean lose decap. The amount of decap that is removed or added to a partition is based on the magnitude of its sensitivity.

Partitions with sensitivities below the mean lose the following percentage of their redistributed decap from their δC_i:

$$\text{Scale}_i = \frac{\zeta\,(S_{\text{mean}} - S_i)}{S_{\text{mean}} - \min(S_{\text{low}})} \tag{10}$$

where Scale_i is the percentage of redistributed decap to remove from the partition, S_{mean} is the mean of the sensitivities, S_i is the sensitivity of partition i, ζ is the maximum fraction of decap that can be redistributed from a single partition at each iteration and S_{low} is a set containing all the sensitivities below the mean.

Partitions with sensitivities above the mean gain the following percentage of the redistributed decap removed from the lower sensitivity partitions to their δC_i:

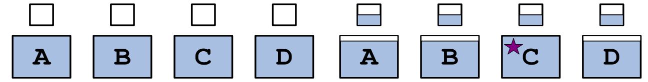
$$\text{Scale}_i = \frac{S_i}{\sum S_{\text{high}}} \tag{11}$$

where S_{high} is a set containing all the sensitivities above the mean.

At each iteration of the algorithm only a fraction (controlled by the variable ζ) of decap from the δC_is is redistributed. A ζ of one leads to the algorithm finishing in one iteration, however the redistribution is only based on the initial sensitivities and consequently such a solution might be far from optimal. Incrementally redistributing the decap obtains better solutions since as the partitions with large sensitivities get more decap their sensitives diminish and other partitions that did not have a high sensitivity become the leading candidate for redistributed decap. The redistribution process is repeated until the change in maximum wire temperature ($T_{\max}$) between iterations is below a threshold or the max number of iterations allowed is reached.

A. Algorithm Example

The algorithm is further illustrated using a simple example. The first stage of the algorithm consists of partitioning the design into blocks (Figure 1(a)) and then calculating the C_{budget} and distributing it evenly to all the blocks (Figure 1(b)). In Figure 1(b), C_{budget} can be interpreted as the sum of all the decap in the δC_is.



(a) Initial circuit is partitioned into 4 blocks A-D. The smaller box above each image represents δC_i for that block.

(b) Decap removed from blocks that do not cause voltage droop is redistributed evenly to the δC_is for all blocks. Note that block C is flagged (Purple star) meaning no decap can be removed from that block.

Fig. 1. Initial decap allocation and redistribution for first stage of the algorithm with decap percentages represented by the size of the shaded area.

Next, the max wire temperature sensitivity with respect to the redistributed decap is calculated for each block which, for this example, is set to $\{0.1, 0.4, 0.7, 0.8\}$. Blocks A and B are then flagged as low sensitivity blocks (mean sensitivity is 0.5) while Blocks C and D are flagged as high sensitivity blocks.

Finally the C_{budget} is redistributed from the δC_is of the low sensitivity blocks to the δC_is of the high sensitivity blocks as illustrated in Figure 2. The amount of redistributed decap removed/added to each block depends on the sensitivity. Block A, has the lowest sensitivity hence the percentage of decap redistributed from Block A is the maximum (ζ) while Block B only loses $\zeta \times \frac{0.5-0.3}{0.5-0.1}$. Block C gains $\frac{0.7}{0.7+0.8}$ of the decap redistributed from Blocks A and B while Block D gains $\frac{0.8}{0.7+0.8}$ of the redistributed decap. The redistribution process is repeated until the stopping criterion is reached.

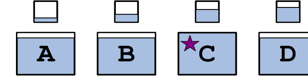


Fig. 2. Decap is moved from the δC_is of the low sensitivity blocks(A and B) to the δC_is of the high sensitivity blocks(C and D)

IV. Experimental Setup

We implemented the prior algorithms in C++. For thermal analysis, we use HotSpot 5.0 [12] using the default parameters and grid mode for more accurate thermal simulations. The direct solver CHOLMOD (Cholesky factorization) from the UFsparse matrix packages [1] is used for transient power grid analysis. The transient solver is implemented using the Backward Euler with a time step of 5ps. Our results are run on a Ubuntu 10.04 Linux system with a 3.4GHz Intel i7-2600 processor and 8GB of memory.

For our experiments we use the IBM power grid transient benchmarks [2]. These PSN benchmarks are extracted from industry designs of IBM circuits. The first two benchmarks are used since they contain the largest total power and are representative of circuits that have significant Joule heating. Since no dimensions are given in the benchmarks, we scaled them so that each chip has an average power density of 250W/cm^2 [10]. The temperature map for each benchmark is computed by partitioning each benchmark into blocks. The blocks are extracted from the benchmark based on current source values. The total current for each block is summed from the current sources within the block and the RMS value is used to calculate the power. The value of δ_{add} for decap redistribution is set to 1%.

The parameters from a 45nm technology were used to calculate the temperature of the wires [6]. The width of the intermediate wires and global wires are set to 70nm and 100nm, respectively. K_{eff} was set to an average of 5Wm^{-1}K^{-1}. The inter-layer dielectric thickness is set to 110nm for the intermediate layers and 215nm for the global layers. The thermal spreading factor (ϕ) is set to 0.88 [3]. The activation energy (Q) value used for MTTF comparisons is set to 0.5eV [3].

V. EXPERIMENTS

We performed several experiments to demonstrate the effectiveness of our decap redistribution algorithms. The first experiment (Section V-A) confirms the large wire temperature increases when decap placement doesn't consider Joule heating. The second experiment (Section V-B) evaluates the effectiveness of our temperature-aware decap redistribution at reducing interconnect temperatures.

A. Baseline Experiment

TABLE I

JOULE HEATING REDUCES THE PSN INTERCONNECT ELECTROMIGRATION LIFETIME BY UP TO $0.12\times$.

Bench	J_w (W)	Max T (K)	Avg ΔT (K)	Max ΔT (K)	ΔR ($\times$)	ΔMTTF ($\times$)
ibmpg1t	2.26	423.1	6.5	56.4	1.2	**0.12**
ibmpg2t	1.97	374.7	1.1	9.0	1.0	**0.68**

The results of the baseline experiment highlighting the effects of Joule heating on PSN interconnect for the IBM transient benchmarks are shown in Table I. The J_w column represents the sum of all the Joule heating RMS values for the wires within the benchmark. The Max T column represents the largest wire temperature. The Avg ΔT and Max ΔT columns represent the average and maximum change in wire temperatures caused by Joule heating respectively. Finally the ΔR and ΔMTTF columns represent the maximum increase in resistivity and maximum decrease in the electromigration lifetime respectively. The large values for ΔR and ΔMTTF especially for the ibmpg1 benchmark clearly shows that Joule heating in PSN interconnect can severely affect the reliability and robustness.

It should be noted that while the total Joule heating RMS power is small compared to the total chip power (about 30W for both benchmarks), it is significant since the area for thermal diffusion (cross-sectional area of wires) is small. The large values for Max ΔT across both benchmark demonstrate the significance of the Joule heating power.

B. Decap Redistribution to Minimize ΔT

We demonstrate the effectiveness of our gradient based thermal aware decap redistribution algorithm in reducing interconnect temperatures and subsequent reliability problems in the IBM transient benchmarks. The results of this experiment depicted in Table II show that the interconnect temperature increases can be greatly reduced by smart decap redistribution leading to increased reliability as measured in electromigration lifetime failure rate. Figure 3 shows an example of how the decap distribution was able to reduce the final wire temperatures for the ibmpg1 benchmark.

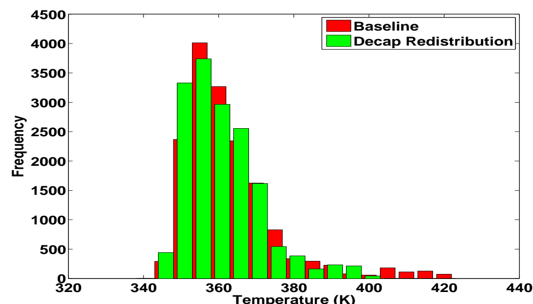


Fig. 3. Histogram of the wire temperatures in the ibmpg1t benchmark illustrating that Decap redistribution can significantly reduce wire temperatures.

TABLE II

TEMPERATURE-AWARE DECAP REDISTRIBUTION INCREASES INTERCONNECT ELECTROMIGRATION RELIABILITY BY A MEAN OF $1.66\times$.

Bench	J_w (W)	Max T (K)	Max ΔT (K)	ΔR ($\times$)	ΔMTTF ($\times$)	Time (sec)
ibmpg1t	1.79	399.8	31.9	0.94	2.24	124
ibmpg2t	1.81	375.0	7.1	0.99	1.08	1679
Improv.	**14.5%**	**12.5K**	**32.2%**	**0.97$\times$**	**1.66$\times$**	

VI. CONCLUSION

In this paper we present a methodology of reducing Joule heating in PSN interconnect by redistributing decap using a gradient based method. Experiments show that our algorithm is able to reduce interconnect temperatures on average by 12.5K which results in a decrease in resistivity by a factor of $0.97\times$ and an increase of electromigration lifetime by a factor of $1.66\times$. We extend our algorithm to consider placing 10% additional decap which results in reduced interconnect temperatures of 19.1K corresponding to a a decrease in resistivity by a factor of $0.96\times$ and increase in electromigration lifetime by a factor of $2.20\times$. In the future we would like to extend our methods to consider 3D-ICs which have PSN wires that are located far away from the substrate and consequently are more prone to Joule heating.

REFERENCES

[1] http://www.cise.ufl.edu/research/sparse/.

[2] http://dropzone.tamu.edu/~pli/PGBench/.

[3] K. Banerjee and A. Mehrotra. Global (interconnect) warming. *Circuits and Devices Magazine, IEEE*, 17(5):16–32, Sep 2001.

[4] T.-Y. Chiang, B. Shieh, and K. Saraswat. Impact of joule heating on scaling of deep sub-micron cu/low-k interconnects. In *Symposium on VLSI Technology*, pages 38–39, 2002.

[5] S. Gurrum, S. Suman, Y. Joshi, and A. Fedorov. Thermal issues in next-generation integrated circuits. *IEEE Transactions on Device and Materials Reliability*, 4(4):709–714, Dec 2004.

[6] S. Im, N. Srivastava, K. Banerjee, and K. Goodson. Scaling analysis of multilevel interconnect temperatures for high-performance ICs. *IEEE Transactions on Electron Devices*, 52(12):2710–2719, Dec 2005.

[7] L. Jiang, Y. Cheng, and J. Mao. Analysis and optimization of thermal-driven global interconnects in nanometer design. *IEEE Transactions on Components, Packaging and Manufacturing Technology*, 1(10):1564–1572, Oct 2011.

[8] A. B. Kahng, B. Liu, and S. X.-D. Tan. Efficient decoupling capacitor planning via convex programming methods. In *ISPD 2006*, pages 102–107, 2006.

[9] H. Li, Z. Qi, S. X.-D. Tan, L. Wu, Y. Cai, and X. Hong. Partitioning-based approach to fast on-chip decap budgeting and minimization. In *DAC 2005*, pages 170–175, 2005.

[10] G. M. Link and N. Vijaykrishnan. Thermal trends in emerging technologies. In *ISQED*, pages 625–632, 2006.

[11] C.-H. Lu, H.-M. Chen, and C.-N. J. Liu. Effective decap insertion in area-array SOC floorplan design. *TODAES*, pages 1–20, 2008.

[12] M. Stan et al. Hotspot: A dynamic compact thermal model at the processor-architecture level. *Microelectronics Journal*, pages 1153–1165, 2003.

[13] H. Su, S. S. Sapatnekar, and S. R. Nassif. An algorithm for optimal decoupling capacitor sizing and placement for standard cell layouts. In *ISPD*, pages 68–73, 2002.

[14] K. Toshiki, O. Takaaki, F. Katsuhiro, T. Hiroshi, K. Atsushi, H. Koutaro, S. Tsuyoshi, T. Masakazu, N. Hidenari, M. Hiroo, S. Takashi, and H. Masanori. Impact of self-heating in wire interconnection on timing. *IEICE Transactions on Electronics*, 93(3):388–392, March 2010.

[15] T.-Y. Wang, J.-L. Tsai, and C. C.-P. Chen. Thermal and power integrity based power/ground networks optimization. In *DATE*, pages 830–835, 2004.

[16] S. Yokogawa, H. Tsuchiya, and Y. Kakuhara. Effective thermal characteristics to suppress joule heating impacts on electromigration in cu/low-k interconnects. In *International Reliability Physics Symposium*, pages 717–723, May 2010.