

Tuning-Friendly Body Bias Clustering for Compensating Random Variability in Subthreshold Circuits

Koichi Hamamoto^{†‡} Masanori Hashimoto^{†‡} Yukio Mitsuyama^{†‡} Takao Onoye^{†‡}

[†]Dept. Information Systems Engineering, Osaka University [‡]JST, CREST
hasimoto@ist.osaka-u.ac.jp

ABSTRACT

Post-fabrication tuning for mitigating manufacturing variability is receiving a significant attention. To reduce leakage increase involved in performance compensation by body biasing, body bias clustering methods have been proposed. However, conventional methods suffer from a large test cost for tuning after fabrication, since there are a tremendous number of body bias assignments. We in this paper propose a low-cost tuning scheme after fabrication and present a layout aware body bias clustering method. The proposed method estimates average leakage power after post-fabrication tuning, and minimizes it. We applied the proposed method to ultra-low voltage circuits for suppressing their high sensitivity to random V_{th} variability, and demonstrated the effectiveness of the proposed method. In the experiments, by just introducing two clusters, leakage power after post-fabrication tuning was reduced by up to 70% compared to a single cluster case.

Categories and Subject Descriptors

B.7 [Integrated Circuits]: Design Aids; B.7 [Integrated Circuits]: Reliability and Testing

General Terms

Algorithms, Design

Keywords

Body Bias Clustering, Performance Compensation, Manufacturing Variability, Layout, Subthreshold Circuits

1. INTRODUCTION

As semiconductor technology advances, manufacturing variability has become one of primary concerns in circuit design, and design for variability has been studied. However, robustness improvement only in design time has its limit, and improvement in performance (timing and power dissipation) and/or yield by design-time optimization is becoming insufficient in advanced technologies. With these backgrounds, performance compensation after

fabrication is gaining popularity, and has been studied. There are mainly two ways to control performance; body biasing and supply voltage scaling. In this paper, we assume that performance is compensated by body biasing, and both forward bias and reverse bias are considered.

An appropriate granularity of performance compensation should be selected according to a dominant component of manufacturing variability. When die-to-die variability is dominant, chip-level performance compensation is reasonable [1]. When spatial variation within a chip is a concern, block-level tuning is more efficient to compensate the performance [2, 3]. However, in a circuit block, there are gates that have less influence on circuit timing, and then block-level compensation often speed up gates that are not relevant to reducing circuit delay, which results in increase in leakage current. To more efficiently compensate performance with smaller power penalty, gate clustering for body biasing has been proposed [4]. The granularity of gate-cluster compensation is between block-level and gate-level compensations, and it has larger freedom for post-fabrication tuning than block-level compensation.

On the other hand, random variation due to RDF (random dopant fluctuation) and LER (line edge roughness) occupies a large portion in total variability, and the portion is expected to increase further in the future. When the random variability becomes significant, chip-level and block-level performance compensations become less efficient. In this case, gate-level compensation might be the best as long as only performance is considered. However, area overhead to separate wells for body biasing becomes prohibitively large. Thus, gate clustering is supposed to be a practical approach to compensate random variation as well as spatial variation within a chip.

Previous works on gate clustering [4, 5] well demonstrate that better performance tuning is possible than block-level tuning. However, there are some difficulties ahead of a practical use. Reference [5] only focuses on performance tunability, and does not aim at compensation of manufacturing variability. Reference [4] proposes to cluster gates based on statistical information of gate-level performance compensation results under variations. The authors claimed that spatial within-die variation made most of neighboring gates being in the same cluster, however it was not clear how applicable this tendency was under actual manufacturing variability. Another problem of [4] is the cost needed to assign body bias voltage after fabrication. The method in [4] provides a tremendous number of potential assignments, and hence finding a good assignment from them for each fabricated chip could be very difficult, since each trial of assignment needs at-speed test and leakage measurement.

We in this paper propose a gate clustering method for compensating random variation whose tuning cost after fabrication is small. The number of required at-speed tests is at most $(\#clusters)+1$,

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, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

ISLPED '09, August 19–21, 2009, San Francisco, California, USA.
Copyright 2009 ACM 978-1-60558-684-7/09/08 ...\$5.00.

and leakage current measurement is not necessary. The proposed method preserves relative gate positions before and after clustering, and hence performance fluctuation after layout realization is limited. As an example of a circuit that is very sensitive to random variation, we focus on subthreshold circuits. To efficiently estimate delay distribution of subthreshold circuits, we develop a statistical timing analysis that handles lognormal distribution of gate delay. To put subthreshold circuits to practical use, overcoming random variability is one of challenges, and this work contributes to it. Die-to-die variation is supposed to be compensated by chip-level compensation such as voltage scaling in this paper.

The rest of this paper is organized as follows. Section 2 explains our approach to simplify post-fabrication tuning. Section 3 describes the proposed clustering method. Section 4 explains leakage estimation after post-fabrication tuning, and Section 5 introduces a statistical timing analysis for subthreshold circuits. Experimental results are shown in Section 6, and the paper is concluded in Section 7.

2. TEST-FRIENDLY POST-FABRICATION TUNING

Conventional body bias clustering [4] assumes that fine tuning after fabrication is possible in testing. Changing body bias of each cluster, an assignment that satisfies timing specification and minimizes leakage power is searched for. In this scheme, fine body bias generation and leakage current measurement in addition to delay testing are necessary. Let the number of possible body bias voltages be M and the number of clusters be N . There are M^N assignments, and then it is difficult to find an optimal assignment via testing.

To overcome the problem above, we propose a body biasing scheme that requires small testing cost for performance compensation. Suppose only two body bias voltages, $V_{b,high}$ and $V_{b,low}$ are available in this work, and $V_{b,high}$ achieves high-speed yet consumes large leakage power, though more voltages might be taken into account by extending the proposed scheme. Figure 1 illustrates how to determine the body bias assignment in the proposed scheme. First, body bias voltages of all clusters ($C_1, C_2, C_3, \dots, C_N$) are assigned to be $V_{b,low}$, which corresponds to performance compensation level A_0 . In this condition, we test whether given timing specification is satisfied. If the specification is not met, the body bias of cluster C_1 is changed to $V_{b,high}$, which corresponds to performance compensation level A_1 . In following, until the timing specification is satisfied, the body bias voltages of clusters C_2, C_3, \dots, C_N are additionally switched to $V_{b,high}$ one by one in the sequence of the predefined cluster number (1 to N).

The advantage of this scheme is that the number of required delay testing is at most $N + 1$. The number of assignments is reduced from M^N to $N + 1$, and the tuning cost after fabrication is significantly reduced. The leakage current monotonically increases as the performance compensation level increments, which makes leakage current measurement unnecessary. In addition, this monotonic level allocation enables dynamic performance compensation after shipping against environmental fluctuation, e.g. temperature, and aging [6].

In the proposed scheme, minimizing leakage current of fabricated chips should be minimizing the average leakage current of chips whose performances are compensated after fabrication, P_{tuning} . Leakage current minimization focusing on a single performance compensation level is not sufficient to minimize the leakage of performance-compensated chips. We thus define the optimization problem as follows.

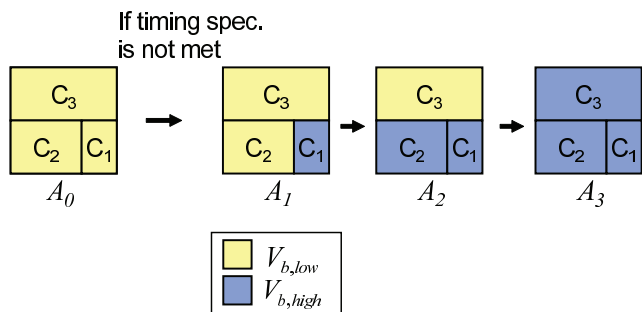


Figure 1: Proposed post-fabrication tuning. #clusters is three in this example.

Minimize:

$$P_{tuning} = \sum_{i=0}^N \{\text{Prob}(A_i) \times P_{avg}(A_i)\}, \quad (1)$$

Subject to:

$$\sum_{i=0}^N \text{Prob}(A_i) \geq Y_{target}, \quad (2)$$

where $\text{Prob}(A_i)$ is the probability that performance compensation level A_i is selected after tuning, $P_{avg}(A_i)$ is the average leakage current at A_i , and Y_{target} is the required yield after compensation. $\text{Prob}(A_i)$ depends on the speed specification. The cost function of Eq. (1) explicitly computes the average leakage current after post-fabrication tuning. The constraint of Eq. (2) is given so that the performance compensation by selecting a level from A_0 to A_N can satisfy the given yield constraint Y_{target} . In this paper, the number of clusters N is assumed to be given, although determining optimal N is also an interesting problem. The computation of Eqs. (1) and (2) will be explained in Section 4.

3. PROPOSED BODY BIAS CLUSTERING

The proposed clustering preserves relative cell placement which is given as an initial layout, and gives a clustering result that minimizes Eq. (1). Figure 2 depicts the procedure of the proposed clustering. An initial layout is assumed to be given.

3.1 Step 1

We first divide the layout into several rectangle regions, and consider these regions as fundamental elements of clustering in the following. We next give an initial solution. Here, a solution is an assignment that each region belongs to one of the clusters (C_1, C_2, \dots, C_N), where N is the number of clusters. In this work, we determine the initial solution by assigning each region to one of the clusters randomly or uniformly for simplicity while another approach, such as based on timing slack, would be possible.

3.2 Step 2

In this step, we search an assignment that minimizes the average leakage after post-fabrication tuning P_{tuning} in Eq. (1). To explore the solution space, we adopted simulated annealing in this work. In the optimization, we generate a neighboring solution in the following two steps; (1) to select a region randomly, (2) to change the cluster which the selected region belongs to by randomly choosing a cluster from C_1 to C_N . We compare P_{tuning} of the neighboring solution to that of the current solution. When

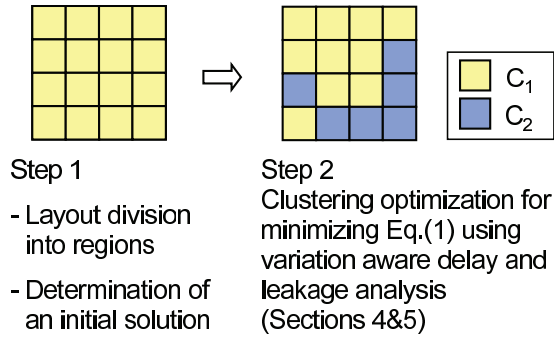


Figure 2: Proposed clustering procedure.

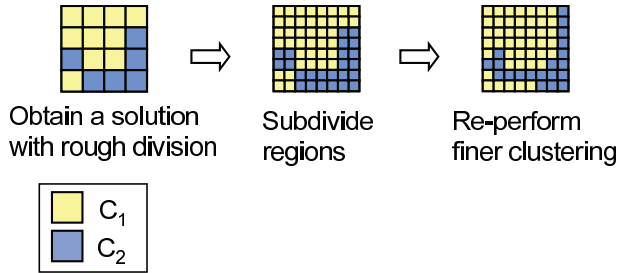


Figure 3: Subdivision of regions and re-execution of finer clustering.

P_{tuning} decreases, the neighboring solution is adopted as a new current solution. When P_{tuning} increases, the neighboring solution is adopted based on the probability $\exp\left(\frac{-\Delta}{T}\right)$, where Δ is the increase in P_{tuning} and T is the temperature parameter. Parameters of initial temperature, annealing schedule and so on are empirically determined considering the obtainable solution quality and CPU time.

When the number of regions is large, the solution space exploration requires longer CPU time. A possible approach to cope with this issue is subdivision of regions after a coarse solution is obtained (Fig. 3). Then, finer clustering is performed within a limited solution space around the previous solution, which may enable speed-up of optimization. This approach trades optimality for scalability.

4. AVERAGE LEAKAGE ESTIMATION AFTER POST-FABRICATION TUNING

This section presents the calculation of the average leakage after post-fabrication tuning P_{tuning} in Eq. (1). We explain the computation of $\text{Prob}(A_i)$ and $P_{avg}(A_i)$ separately.

4.1 Computation of $\text{Prob}(A_i)$

The probability that a performance compensation level A_i is selected can be computed using cumulative distribution function (CDF) of circuit delay at A_i , D_i , and delay constraint D_C . When the circuit delay is estimated with an approximation to Gaussian distribution, which is a popular approximation found in literatures[7], the probability that the delay constraint D_C is satisfied

at level A_i , $\text{Prob}(D_i < D_C)$ is expressed as

$$\text{Prob}(D_i \leq D_C) = \Phi\left(\frac{D_C - E(D_i)}{\sqrt{V(D_i)}}\right), \quad (3)$$

where Φ is the cumulative distribution function of the standard Gaussian distribution, E is the average, and V is the variance.

$$\Phi(z) = \int_{-\infty}^z \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{x^2}{2}\right) dx. \quad (4)$$

Assuming that delay testing is perfectly performed after fabrication, $\text{Prob}(A_i)$ can be computed as

$$\text{Prob}(A_i) = \begin{cases} \text{Prob}(D_i \leq D_C) & (i = 0), \\ \text{Prob}(D_i \leq D_C) - \sum_{k=0}^{i-1} \text{Prob}(A_k) & (i > 0). \end{cases} \quad (5)$$

Generally, CDF of D_i is obtained by SSTA (statistical static timing analysis) or Monte Carlo simulation. When computing $\text{Prob}(D_i \leq D_C)$ at a different compensation level, we update gate delay parameters, such as average and standard deviation, corresponding to the compensation level, execute SSTA or Monte Carlo simulation, and calculate Eq. (3). SSTA tailored for subthreshold circuit will be presented in Section 5.

4.2 Computation of $P_{avg}(A_i)$

We next explain the computation of the average leakage at level A_i . Subthreshold leakage is sensitive to threshold voltage (V_{th}) variation, and hence we here focus on subthreshold leakage current, and leakage power dissipation of gate j is modeled as

$$p_j = a_j \exp(b_j X_j), \quad (6)$$

where a_j and b_j are fitting parameters and determined by fitting to circuit simulation results. a_j and b_j are derived for each cell at each body bias voltage. X_j is V_{th} variable of gate j and given as a Gaussian distribution with average μ_j and variance σ_j^2 . V_{th} variation mostly originates from random dopant fluctuation, and hence we assume that X_j and X_k ($j \neq k$) are uncorrelated.

Given a gate leakage model in Eq. (6), $E(P)$, average of leakage current P is expressed as Eq. (7) referring to [8].

$$E(P) = \sum_{j=1}^{N_{gate}} \left\{ a_j \exp\left(b_j \mu_j + \frac{b_j^2 \sigma_j^2}{2}\right) \right\}, \quad (7)$$

where N_{gate} is the number of gates in the circuit. By computing $E(P)$ using appropriate a_j and b_j for each gate based on the body bias voltages of the cluster assignment at A_i , we can obtain $P_{avg}(A_i)$.

4.3 Correlation between leakage and delay

In the previous section, $P_{avg}(A_i)$ is computed by averaging leakage power in the whole variation parameter space Ω . However, strictly speaking, performance compensation level A_i is selected only in a sub-space of variation parameters, which means $P_{avg}(A_i)$ in the previous section could be inaccurate.

Let Ω_i ($i = 0, 1, \dots, N$) denote the sub-space of variation parameters so that the performance compensation level A_i is selected.

$$\Omega = \cup_{i=0}^N \Omega_i + \Omega_{out}, \quad (8)$$

where Ω_{out} corresponds to the sub-space that the timing specification is not satisfied even at level A_N .

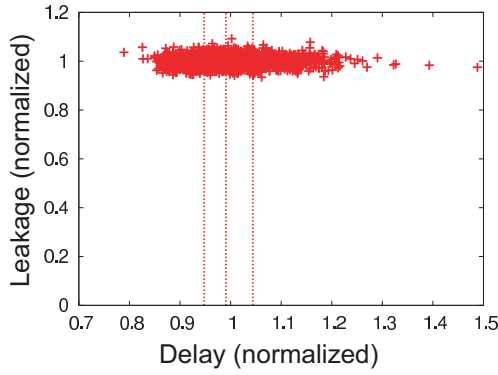


Figure 4: Relation between delay and leakage power in a 16-bit multiplier under random Vth variation (vertical dotted lines represent quartiles).

Letting $P_{avg}(A_i; \Omega_i)$ denote the average leakage current at level A_i in sub-space Ω_i , P_{tuning} should be expressed as

$$P_{tuning} = \sum_{i=0}^N \{Prob(A_i) \times P_{avg}(A_i; \Omega_i)\}. \quad (9)$$

However, $P_{avg}(A_i; \Omega_i)$ is difficult to compute, since both the identification of Ω_i and the integration of P_{avg} in Ω_i are not easy.

We then deliberated the situation when $P_{avg}(A_i; \Omega_i)$ and $P_{avg}(A_i; \Omega)$ become different. The difference becomes significant in the case that delay and leakage power are correlated. Focusing on die-to-die variability, they are absolutely correlated. However, this work aims to combat the random variability, assuming that the die-to-die variability is compensated by, for example, Vdd adjustment. We thus evaluate the correlation between delay and leakage under random Vth variation.

Figure 4 shows the relation between delay and leakage of a 16-bit multiplier obtained by Monte Carlo simulation. Similarly to experiments in Section 6, subthreshold operation at $V_{dd} = 300$ mV was assumed. We gave random Vth variation whose standard deviation was 25 mV. The evaluation count is 2,000. For each trial, we computed the circuit delay and leakage power. The vertical dotted lines correspond to 25%, 50% and 75% tile values. The average leakage in each quartile range is computed and listed in Table 1. We can see that the average leakages in four ranges are almost identical and virtually equal to the average leakage in the entire range. In fact, the correlation coefficient between delay and leakage is 0.0064, and they are not correlated. We therefore conclude that $P_{avg}(A_i; \Omega_i)$ and $P_{avg}(A_i; \Omega)$ are almost identical, and $P_{avg}(A_i; \Omega_i)$ can be replaced by $P_{avg}(A_i; \Omega)$, as long as only random Vth variation is considered.

Table 1: Average leakage current in each delay quartile (16-bit multiplier).

Delay	#samples	Average leakage (relative)
0-25% tile	499	0.9987
25-50% tile	503	1.0000
50-75% tile	497	1.0001
75-100% tile	501	0.9993
Total	2000	1

5. STATISTICAL TIMING ANALYSIS FOR SUBTHRESHOLD CIRCUITS

When supply voltage is close to subthreshold voltage, drain current changes exponentially according to Vth variation, and hence linear approximation of delay variation due to Vth fluctuation is inappropriate. We thus model gate delay of gate j , d_j , is expressed as

$$d_j = a_j \exp(b_j X_j), \quad (10)$$

where a_j and b_j are fitting parameters and X_j is Vth variable of gate j . X_j and X_k ($j \neq k$) are assumed to be uncorrelated.

The average and variance of gate delay d_j are calculated by

$$E(d_j) = a_j \exp\left(b_j \mu_j + \frac{b_j^2 \sigma_j^2}{2}\right), \quad (11)$$

$$V(d_j) = a_j^2 \exp(2b_j \mu_j + b_j^2 \sigma_j^2) \{\exp(b_j^2 \sigma_j^2) - 1\}. \quad (12)$$

To compute circuit delay, we need two operations; sum operation of signal arrival time and gate delay, and maximum operation of two signal arrival times. The sum of signal arrival time AT and gate delay d is given by

$$E(AT + d) = E(AT) + E(d), \quad (13)$$

$$V(AT + d) = V(AT) + V(d) + 2Cov(AT, d), \quad (14)$$

where $Cov(AT, d)$ is the covariance between AT and d and is zero in this work since uncorrelated random variability is considered.

As for the maximum operation for AT_j and AT_k , we regard the distributions of AT_i and AT_j as lognormal distributions [9], and perform max operation tailored for lognormal distributions [10]. Consequently, the average and variance are expressed as

$$E(\max(AT_i, AT_j)) = E(AT_i) \Phi \left[\frac{m_i - m_j + v_i}{\sqrt{v_i + v_j}} \right] + E(AT_j) \Phi \left[\frac{-(m_i - m_j) + v_j}{\sqrt{v_i + v_j}} \right], \quad (15)$$

$$V(\max(AT_i, AT_j)) = E(AT_i^2) \Phi \left[\frac{m_i - m_j + 2v_i}{\sqrt{v_i + v_j}} \right] + E(AT_j^2) \Phi \left[\frac{-(m_i - m_j) + 2v_j}{\sqrt{v_i + v_j}} \right] - E^2(\max(AT_i, AT_j)). \quad (16)$$

Here, $\Phi(z)$ is the cumulative distribution function of the standard Gaussian distribution in Eq. (4), and m_i, m_j, v_i and v_j are given by

$$\begin{aligned} m_i &= \frac{1}{2} \log \left(\frac{E^4(AT_i)}{E^2(AT_i) + V(AT_i)} \right), \\ m_j &= \frac{1}{2} \log \left(\frac{E^4(AT_j)}{E^2(AT_j) + V(AT_j)} \right), \\ v_i &= \log \left(\frac{V(AT_i) + E^2(AT_i)}{E^2(AT_i)} \right), \\ v_j &= \log \left(\frac{V(AT_j) + E^2(AT_j)}{E^2(AT_j)} \right). \end{aligned} \quad (17)$$

We thus perform SSTA approximating the distributions of signal arrival time to lognormal distribution, and obtain the information needed to compute $Prob(A_i)$.

Table 2: Circuits used for experiments.

Circuit	#cells	Delay constraint[ns]
16-bit multiplier (mult16)	3,987	2,350
ALU	10,611	12,000
32-bit multiplier (mult32)	14,685	3,000
64-bit multiplier (mult64)	70,595	3,850

6. EXPERIMENTAL RESULTS

We experimentally evaluate the effectiveness of the proposed body bias clustering. Circuits used for experiments are designed using an industrial 65nm library (Table 2). We assume that the standard deviation of V_{th} random variation is 25 mV. The supply voltage is 300 mV and the number of clusters is two. We suppose that body bias is given by swapped body bias (SBB) [11], since it is area efficient and suitable for low-voltage circuit design. The target yield Y_{target} in Eq. (2) is 99.7%. The proposed method is implemented in C++ language and the program was run on a 2.4 GHz Opteron processor.

We compare the average leakage current after performance compensation without clustering to that with clustering. The result is shown in Fig. 5. We performed clustering in two configurations of 4×4 regions and 8×8 regions. With body bias clustering, the average leakage current is reduced by up to 70%. By clustering gates with careful consideration of timing, the proposed method makes only a small part of the circuit forward biased, whereas the compensation without clustering enforces the entire circuit forward biased. Therefore, the proposed method achieves smaller leakage current. Spatial division of 8×8 regions further reduces the leakage than that of 4×4 regions. This is reasonable, since the solution space is expanded and finer clustering becomes possible.

We next evaluate the relation between the given delay constraint and the average of leakage power dissipation. Figure 6 shows the result in a 16-bit multiplier. Without body bias clustering, when the delay constraint becomes tighter from 2500 ns to 2350 ns, almost all chips are forward biased, and the average leakage increases rapidly. The average leakage is constant below 2350 ns, since all chips are forward biased under these constraints. In contrast, the proposed method gradually increases the size of cluster C_1 to meet the delay constraint, and hence the increase in average leakage is moderate.

Figure 7 shows the relation between circuit size and CPU time, when the number of iterations in SA was set unchanged. This CPU time includes times for reading netlist and output a result in addition to clustering optimization with SSTA. Figure 7 indicates that CPU time is proportional to the number of instance. This is because the computation complexity of SSTA used in the proposed method is $O(N_g + N_i)$ [12], where N_g and N_i are the numbers of gates and interconnects. In the current implementation, incremental timing update, which is commonly used in timing analysis inside optimization loops, was not implemented. Thus further speed up would be possible.

We finally arrange the initial layout so that different body biasing becomes possible by inserting deep N-well separation. We first insert deep N-well separation between different clusters. At this moment, the layout is not rectangle. We thus perform ECO (engineering change order) placement using a commercial P&R tool so that the perturbation from the initial layout is minimized, and obtain a rectangle layout. The impact of ECO placement on timing is limited, since the delay fluctuation due to manufacturing variability is much larger and the impact is overwhelmed by the variability.

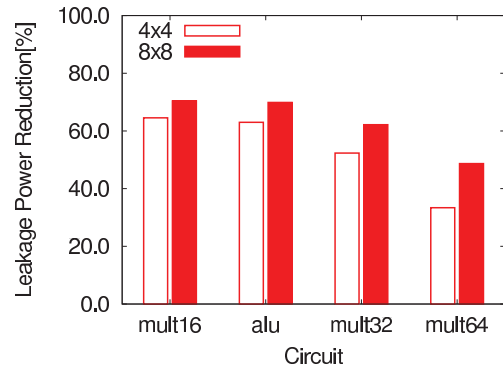
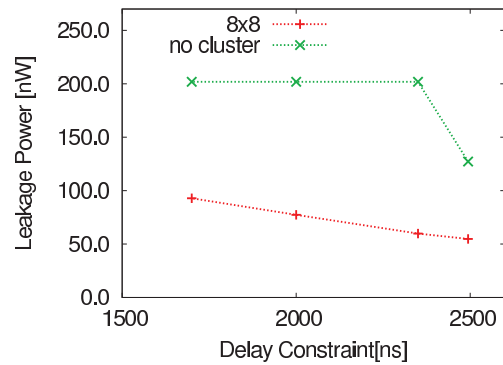
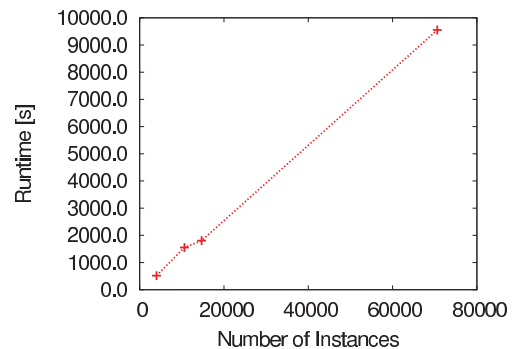
**Figure 5: Leakage power reduction by clustering.****Figure 6: Delay constraint and average leakage power after post-fabrication tuning.****Figure 7: Relation between circuit size and CPU time.**

Figure 8 shows an example of the layout after inserting separation. The smaller cluster is cluster C_1 , and the other is C_2 . The white area is the inserted separation. In this example of 16-bit multiplier, the area overhead due to separation is 8.6%. The overhead depends on the circuit size, the number of clusters and the number of regions divided into in the clustering procedure as well as technology design rules. We thus have to appropriately choose the numbers of regions and clusters taking the trade-off between leakage reduction and area overhead into consideration.

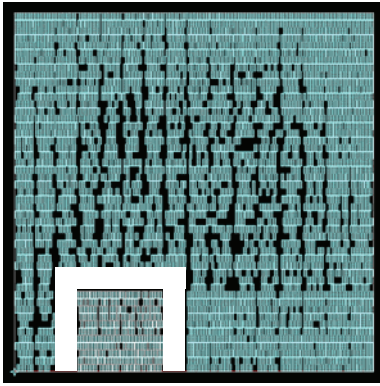


Figure 8: An example of layout after clustering (16-bit multiplier).

7. CONCLUSION

We proposed a layout-aware clustering method for body biasing whose tuning cost after fabrication is small, and applied it to subthreshold circuits. We devised a leakage estimation method after performance compensation and a statistical timing analysis method for subthreshold circuits. Using these methods, the proposed method minimizes the leakage current after timing requirement is satisfied by post-fabrication tuning. In experiments, the proposed method was applied to four circuits under various timing constraints to compensate random V_{th} variability. The average leakage after tuning was reduced by up to 70%, and an layout realization after clustering was demonstrated.

Acknowledgment

This work was partly supported by NEDO.

8. REFERENCES

- [1] M. Takahashi, M. Hamada, T. Nishikawa, H. Arakida, T. Fujita, F. Hatori, S. Mita, K. Suzuki, A. Chiba, T. Terazawa, F. Sano, Y. Watanabe, K. Usami, M. Igarashi, T. Ishikawa, M. Kanazawa, T. Kuroda and T. Furuyama, "A 60-mW MPEG4 Video Codec Using Clustered Voltage Scaling with Variable Supply-Voltage Scheme," *IEEE Journal of Solid-State Circuits*, Vol. 33, No. 11, pp. 1772-1780, Nov. 1998.
- [2] J. W. Tschanz, J. T. Kao, S. G. Narendra, R. Nair, D. A. Antoniadis, A. P. Chandrakasan and V. De, "Adaptive Body Bias for Reducing Impacts of Die-to-Die and Within-Die Parameter Variations on Microprocessor Frequency and Leakage," *IEEE Journal of Solid-State Circuits*, vol. 37, No. 11, pp. 1396–1402, Nov. 2002.
- [3] Y. Nakamura, D. Levacq, L. Xiao, T. Minakawa, T. Niiyama, M. Takamiya, and T. Sakurai, "1/5 Power Reduction by Global Optimization based on Fine-Grained Body Biasing", in *Proc. CICC* pp. 547–550, 2008.
- [4] S. H. Kulkarni, D. M. Sylvester and D. T. Blaauw, "Design-Time Optimization of Post-Silicon Tuned Circuits Using Adaptive Body Bias," *IEEE Trans. on CAD*, vol. 27, No. 3, pp. 481–494, Mar. 2008.
- [5] V. Khandelwal and A. Srivastava, "Active Mode Leakage Reduction Using Fine-Grained Forward Body Biasing Strategy," in *Proc. ISLPED*, pp. 150–155, 2005.
- [6] H. Fuketa, M. Hashimoto, Y. Mitsuyama and T. Onoyo: "Trade-off Analysis between Timing Error Rate and Power Dissipation for Adaptive Speed Control with Timing Error Prediction," in *Proc. ASP-DAC*, pp. 266–271, 2009.
- [7] D. Blaauw, K. Chopra, A. Srivastava and L. Scheffer, "Statistical Timing Analysis: From basic principles to state-of-the-art," *IEEE Trans. CAD*, Vol. 27, No. 4, pp.589–607, April 2008.
- [8] R. Rao, A. Srivastava, D. Blaauw and D. Sylvester, "Statistical Analysis of Subthreshold Leakage Current for VLSI Circuits," *IEEE Trans. on Very Large Scale Integration Systems*, Vol. 12, No. 2, pp.131–139, Feb. 2004.
- [9] L. Fenton, "The Sum of Log-Normal Probability Distributions in Scatter Transmission Systems," *IRE Trans. on Communications Systems*, vol. 8, pp. 57–67, Mar. 1960.
- [10] D. Lien, "Moments of Ordered Bivariate Log-Normal Distributions," *Economics Letters*, vol. 20, pp. 45–47, 1986.
- [11] S. Narendra, J. Tschanz, J. Hofsheier, B. Bloechel, S. Vangal, Y. Hoskote, et al., "Ultra-Low Voltage Circuits and Processor in 180nm to 90nm Technologies with a Swapped-Body Biasing Technique," in *ISSCC Digest of Technical Papers*, pp. 156–157, 2004.
- [12] H. Chang and S. S. Sapatnekar, "Statistical Timing Analysis Under Spatial Correlations," *IEEE Trans. on CAD*, vol. 24, no. 9, pp. 1467–1482, Sept. 2005.