Comparative study on delay degrading estimation due to NBTI with circuit/instance/transistor-level stress probability consideration

Hiroaki Konoura†*, Yukio Mitsuyama†*, Masanori Hashimoto†*, and Takao Onoye†*
†Dept. Information Systems Engineering, Osaka University, Japan *JST CREST
konoura.hiroaki, mituyama, hasimoto, onoye}@ist.osaka-u.ac.jp

Abstract—NBTI degradation proceeds while a negative bias is applied to the gate of PMOS, whereas it recovers while a positive bias is applied. Therefore, PMOS stress (ON) probability has a strong impact on circuit timing degradation due to NBTI effect. This paper evaluates how the granularity of stress probability calculation affects NBTI prediction using the state-of-the-art long term prediction model. Experimental results show that the prediction accuracy of timing degradation due to NBTI effect is heavily dependent on granularity of stress probability consideration in timing analysis.

I. Introduction

Negative bias temperature instability (NBTI) is one of the serious concerns in nanoscale integrated circuits design. NBTI is the degradation effect which causes gradual increase of ΔV_{th} while a negative bias is applied to PMOS. This condition is defined as stress phase of NBTI. If NBTI stress continues for a long time, path delay increases and may lead to a timing error. On the other hand, while PMOS is OFF, ΔV_{th} gradually decrease to its initial value before stress injection, and PMOS degradation is relaxed. This condition is defined as recovery phase of NBTI. Degradation caused by NBTI increases gradually repeating stress and recovery cycles [1].

The degradation by NBTI depends on operational parameters such as temperature, supply voltage, and stress probability [2, 3]. Stress probability is defined as time ratio of stress phase, that is (time of stress) / (time of stress + time of recovery). When the stress probability is almost 100% and PMOS is under stress for a long time, an increase in path delay becomes extremely large due to large $\mathrm{V_{th}}$ shift.

In order to predict NBTI effect, a long term prediction model has been proposed[1]. For accurate prediction of circuit delay degradation, appropriate stress probability estimation and consideration in timing analysis are crucially important.

Reference [4] proposed a transistor-level estimation method of stress probability. Accurate estimation is possible, however, a large computational cost is required for enormous number of transistors in a large-scale circuit. On the other hand, gate-level estimation of path delay using NBTI aware static timing analysis is proposed in [5]. Thus, there are several proposals with different granularities, however, the importance of the granularity (circuit/instance/transistor) in probability estimation and timing analysis has not been sufficiently discussed.

In this paper, we present case studies of circuit degradation using three granularities of stress probability consideration, that is levels of circuit, instance, and transistor. We evaluate critical path delay, distribution of estimated delay difference for each path, and the order of path delays. We also discuss the relation between granularities of stress probability consideration and performance degradation by NBTI.

The rest of this paper is organized as follows. The prediction model of NBTI[1] used in this work is described in Section II. Section III presents three methods of stress probability computation and procedure of timing degradation analysis. Section IV shows evaluation results of timing degradation and its investigation. Finally, the discussion is concluded in Section V.

II. Prediction model of NBTI effect

Various prediction models of NBTI has been proposed in [1, 6–8]. In this paper, we use a long term prediction model that includes recovery effect and is useful for estimation of degradation by years[1]. In this model, $\mathrm{V_{th}}$ degradation after time t has passed ($\Delta V_{th,t}$) is expressed as Eq. (1).

$$\Delta V_{th,t} = \left(\frac{\sqrt{K_v^2 \cdot T_{clk} \cdot \alpha}}{1 - \beta_t^{1/2n}} \right)^{2n}. \quad (1)$$

In Eq. (1), K_v is a parameter dependent on supply voltage and temperature. T_{clk} is clock period, and α is stress probability of PMOS. β_t is a parameter that has a dependence on temperature, T_{clk}, α, and t. Moreover, n is equal to 1/6 in a hydrogen molecule diffusion based model[9]. All of these parameters are important for the progress of NBTI degradation[10].

When the operating frequency is higher than 10 kHz, there is little relation between $\Delta V_{th,t}$ and T_{clk} [1]. In this case, Eq. (2) is used for $\Delta V_{th,t}$ estimation instead of Eq. (1).

$$\Delta V_{th,t} \approx \left(\frac{0.001 n^2 K_v^2 \alpha C t}{0.81 t_{ox}^2 (1-\alpha)} \right)^{n}. \quad (2)$$

In Eq. (2), parameter C is dependent on temperature, and t_{ox} denotes gate oxide thickness. Here, in Eq. (2), when α approaches to 1, $\Delta V_{th,t}$ reaches an infinite value and is not appropriate. In such a situation, as its upper limit, we use Eq. (3) which models only stress phase of NBTI[11].

$$\Delta V_{th,t} = \left(K_v^2 t\right)^n. \quad (3)$$

Let us show an example of NBTI degradation. Figure 1 shows $\mathrm{V_{th}}$ degradation calculated with Eq. (2), referring a parameter set of 65 nm process in [1]. A significant $\mathrm{V_{th}}$ degradation can be found in the first year. In addition, we can see the dependency of $\mathrm{V_{th}}$ degradation on α. We also applied this $\mathrm{V_{th}}$ degradation to all PMOSs in a small combinational circuit shown in Fig. 2. After ten years, the critical path delay increases by 1.5% ($\alpha = 0.1$), 2.4% ($\alpha = 0.5$), 7.0% ($\alpha = 0.9$), respectively. Thus, increase in path delay depends on α.

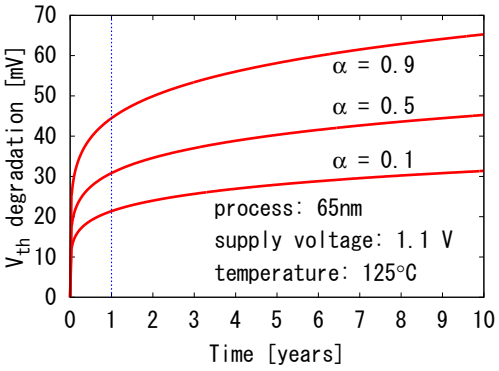


Fig. 1. Threshold voltage degradation predicted with long term model [1] .

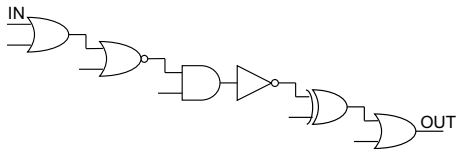


Fig. 2. An example of combinational circuit.

III. Evaluation methods of NBTI effect

This section describes three methods under evaluation of stress probability computation of each PMOS and the procedure of timing degradation analysis.

A. Stress probability computation

We use three stress probability calculation methods (SPCM) with different granularities as follows.

SPCM-A. Set stress probability of all PMOSs to 50% uniformly.

SPCM-B. Set stress probability of all PMOSs in an instance to the state probability of the instance output being high.

SPCM-C. Calculate stress probability for each transistor.

SPCM-A is the simplest method, which is based on an assumption that each part of circuit works uniformly. Reference [12] reported that 50% is a reasonable value to conjecture the maximum timing degradation. In SPCM-B, the stress probability of PMOSs in an instance is assumed to be identical to the state probability of the instance output, which can be obtained by logic simulation. SPCM-B can be performed without taking into consideration the transistor-level structure inside the instance. In SPCM-C, the stress probability of each PMOS is individually calculated by considering the connection of all transistors. Comparing to other methods, SPCM-C is the finest-grained calculation method and is expected to obtain the most accurate stress probability.

In this work, SPCM-C is executed on the basis of logic simulation results, which means the state probabilities of all the nets are given. The computation procedure is:

1. For each cell, examine whether each PMOS is under stress or not for every combination of input states.
2. Calculate the probabilities of all the combinations of input states at each instance by using the state probabilities of the nets. We here call this probability as combinatorial probability.
3. Obtain the stress probability of each PMOS using the above two information.

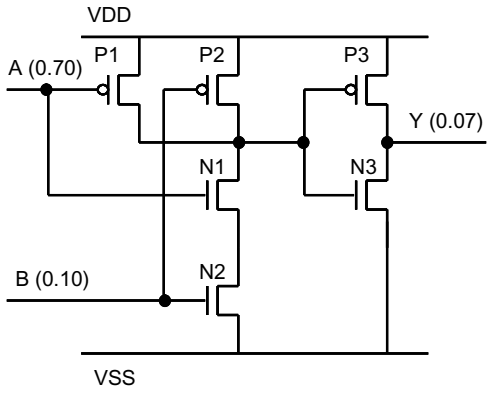


Fig. 3. 2-input AND gate. Values noted at input and output terminals are state probabilities of being 1.

TABLE I
COMBINATORIAL PROBABILITY AND PMOS STRESS CONDITION.

{A, B}	combinatorial probability	P1	P2	P3
{0, 0}	0.30×0.90	stress	stress	-
{0, 1}	0.30×0.10	stress	-	-
{1, 0}	0.70×0.90	-	stress	-
{1, 1}	0.70×0.10	-	-	stress
stress probability		0.30	0.90	0.07

TABLE II
RESULT OF STRESS PROBABILITY ESTIMATION (2-INPUT AND GATE).

stress probability	SPCM-A [%]	SPCM-B [%]	SPCM-C [%]
P1	50.0	7.0	30.0
P2	50.0	7.0	90.0
P3	50.0	7.0	7.0

As an example, we apply SPCM-C to 2-input AND shown in Fig. 3. Supposing the state probabilities of inputs {A, B} are 0.7 and 0.1 respectively, the combinatorial probabilities of {0, 0}, {0, 1}, {1, 0}, and {1, 1} are listed in Table I. Here, correlation between inputs is not considered, though the state probabilities of inputs {A, B} are exactly calculated by logic simulation results. Then, we examine whether P1, P2, and P3 are under stress for each input combination. As shown in Table I, the stress probability of P1 is expressed as a summation of the combinational probabilities, of which P1 is ON, $\alpha = 0.30 \times 0.90 + 0.30 \times 0.10 = 0.30$.

The stress probabilities calculated by each SPCM are shown in Table II. Each SPCM gives different stress probabilities. Focusing on P2, there are large difference, where the stress probabilities are 50.0%, 7.0%, 90.0% in SPCM-A-C, respectively.

B. Procedure of timing degradation analysis

In order to evaluate timing degradation due to NBTI, it is necessary to calculate all PMOS stress probabilities and run static timing analysis applying V_{th} degradation. Timing analysis based on SPCM-A and -B can be performed at gate-level with cell delay characterization that includes an additional parameter V_{th}. SPCM-C, however, requires the largest computational cost of estimating path delay using V_{th} degradation for each PMOS as well as of calculating different stress probability. Instance delay increase must be computed with different V_{th} shifts of each PMOS, which requires dedicated cell delay model such as [13] or transistor-level timing analyzer.

Figure 4 illustrates the overall procedure adopted for this

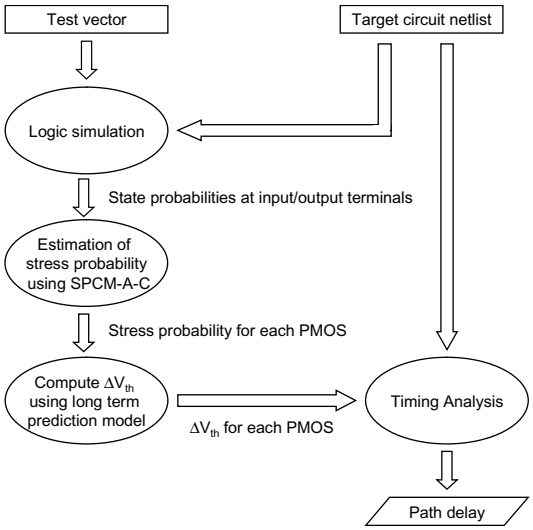


Fig. 4. Estimation procedure of circuit degradation by NBTI.

TABLE III
D-FPU SPECIFICATION.

input/output operand	64 bits
function	four arithmetic operations, rounding
operating frequency	167 MHz
gate count	84,949

TABLE IV
AES SPECIFICATION.

key length	128 bits
block size	128 bits
function	encryption
operating frequency	500 MHz
gate count	24,927

evaluation. Firstly, the state probabilities at input/output terminals of all instances are calculated from logic simulation results using given test vectors. Secondary, the stress probabilities of all PMOSs are calculated by SPCM-A-C. After that, the stress probability for each PMOS is converted to V_{th} degradation referring the of long term model(Eq. (2)). Finally, we perform timing degradation analysis using a commercial transistor-level static timing analyzer (Synopsys PathMill[14]) in this work. The information on V_{th} degradations of all PMOSs are embedded in the transistor-level netlist given to PathMill. In addition to three SPCM, we define the worst case, in which stress probability of all PMOSs is 100%.

IV. Evaluation results of timing degradation

This section shows evaluation results assuming several operating situations and its consideration.

A. Circuits for evaluation

In this paper, double floating point unit (D-FPU) and symmetric-key cipher algorithm AES [15] are adopted as target circuits for evaluation. The specifications and implemented results of D-FPU and AES using an industrial 65 nm standard cell library are summarized in Table III and IV, respectively.

Input and output operands of D-FPU are 64 bits, which consists of 1 bit sign, 11 bits exponent, and 52 bits mantissa. On the other hand, AES performs encryption of 128-bit plaintext with 128-bit key, and outputs 128-bit ciphertext.

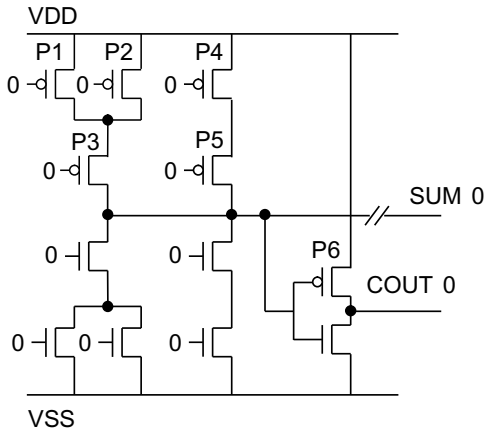


Fig. 5. Full adder (carry out).

B. Evaluation of critical path delay

B.1 Results

Following the procedure in Fig. 4, delay increase of critical path is estimated. We assume three operating situations; (a) general operation, (b) biased operation, and (c) biased data, since stress probability depends on circuit operation. For each situation, 100,000 test vectors were prepared for D-FPU. On the other hand, for AES, 100 keys and 1,000 plain texts were prepared.

B.1.a **General operation.** In this evaluation, both operation and operand are randomly selected and executed. Table V shows the evaluation results of NBTI degradation after 10 years. In D-FPU, we can see only small differences of delay increase among SPCM-A-C. Similarly, in AES, the difference is very small.

B.1.b **Biased operation.** Table VI shows the evaluation results, where D-FPU executes only addition. In Table VI, the amount of delay increase with SPCM-C is larger than those with SPCM-A and -B at higher temperature. Moreover, as the temperature is raised, its difference becomes larger. Comparing with Table V, the amount of delay increase at 125°C in SPCM-C becomes 5.4 times larger.

B.1.c **Biased data.** We here generated test vectors for D-FPU so that operands whose 8-bit of 11-bit exponent and 40-bit of 52-bit significand were fixed to 0. As for AES, 96-bit of 128-bit plain text were fixed to 0. Table VII and VIII show the results of D-FPU and AES respectively. In Table VII, the amounts of delay increase are different among SPCM-A-C. Moreover, in case of D-FPU, compared with Table V, the amount of delay increase is 7.0 times larger than that at 125°C in SPCM-C. As for AES, comparing Table VIII with Table V, the difference between SPCM-A and SPCM-B-C became large.

B.2 Discussion

B.2.a **Impact of inactive circuit portion.** In Table VI, where only additions were performed, in order to examine the difference of critical path delay between SPCM-B and SPCM-C, the critical path estimated with SPCM-C was investigated. We then came to the conclusion that delay greatly increased through a

TABLE V

CRITICAL PATH DELAY DEGRADATION(GENERAL OPERATION, 10 YEARS LATER).

		temperature [°C]	initial	worst	SPCM-A	SPCM-B	SPCM-C
D-FPU	critiacl path delay [ns] (degradation [ns])	25	3.452	4.019 (0.567)	3.508 (0.056)	3.509 (0.057)	3.469 (0.017)
		75	3.458	5.154 (1.696)	3.578 (0.120)	3.663 (0.205)	3.714 (0.256)
		125	3.632	6.093 (2.461)	3.908 (0.276)	3.927 (0.295)	3.895 (0.263)
AES	critical path delay [ns] (degradation [ns])	25	1.096	1.286 (0.190)	1.122 (0.026)	1.122 (0.026)	1.124 (0.028)
		75	1.147	1.550 (0.403)	1.189 (0.042)	1.210 (0.063)	1.204 (0.057)
		125	1.166	1.799 (0.633)	1.260 (0.094)	1.284 (0.118)	1.279 (0.113)

TABLE VI

CRITICAL PATH DELAY DEGRADATION (D-FPU, ONLY ADDITION, 10 YEARS LATER).

	temperature [°C]	initial	worst	SPCM-A	SPCM-B	SPCM-C
critical path delay [ns] (degradation [ns])	25	3.452	4.019 (0.567)	3.508 (0.056)	3.509 (0.057)	3.511 (0.059)
	75	3.458	5.154 (1.696)	3.578 (0.120)	3.588 (0.130)	4.306 (0.848)
	125	3.632	6.093 (2.461)	3.908 (0.276)	3.924 (0.292)	5.050 (1.418)

TABLE VII

CRITICAL PATH DELAY DEGRADATION (D-FPU, GENERAL OPERATION, USE BIASED DATA, 10 YEARS LATER).

	temperature [°C]	initial	worst	SPCM-A	SPCM-B	SPCM-C
critical path delay [ns] (degradation [ns])	25	3.452	4.019 (0.567)	3.508 (0.056)	3.507 (0.055)	3.810 (0.358)
	75	3.458	5.154 (1.696)	3.578 (0.120)	3.607 (0.149)	4.665 (1.207)
	125	3.632	6.093 (2.461)	3.908 (0.276)	3.906 (0.274)	5.466 (1.834)

TABLE VIII

CRITICAL PATH DELAY DEGRADATION (AES, USE BIASED PLAIN TEXT, 10 YEARS LATER).

	temperature [°C]	initial	worst	SPCM-A	SPCM-B	SPCM-C
critical path delay [ns] (degradation [ns])	25	1.096	1.286 (0.190)	1.122 (0.026)	1.158 (0.062)	1.151 (0.055)
	75	1.147	1.550 (0.403)	1.189 (0.042)	1.273 (0.126)	1.258 (0.111)
	125	1.166	1.799 (0.633)	1.260 (0.094)	1.389 (0.223)	1.347 (0.181)

number of carrying part of full adders, which are used for multiplication. In biased data case of Table VII, focusing on the critical path of SPCM-C, a similar tendency was seen through full adders which are used for addition.

Figure 5 shows a circuit schematic of carry generation part of a full adder. In Fig. 5, all input and output terminals are constantly 0. In this case, each SPCM estimates stress probabilities as listed in Table IX. The estimated stress probabilities are totally different. When the inputs of full adder are constantly 0 as shown in Fig. 5, the outputs are also 0, and hence, in SPCM-B, all stress probabilities are settled to be 0%.

Meanwhile, Fig. 6 shows correspondence of $\mathrm{V_{th}}$ degradation to various stress probability assuming 125°C and 10-year operation, using Eqs. (2) and (3). The stress probability difference between 0% and 50% causes 50 mV $\mathrm{V_{th}}$ difference. On the other hand, the difference between 50% and 100% induces 150 mV shift indeed, and is three times larger. When the stress probability approaches 100%, $\mathrm{V_{th}}$ degradation drastically increases. With SPCM-B, since the number of PMOSs whose stress probabilities are 100% are fewer than SPCM-C, the path delay degradation becomes smaller.

TABLE IX

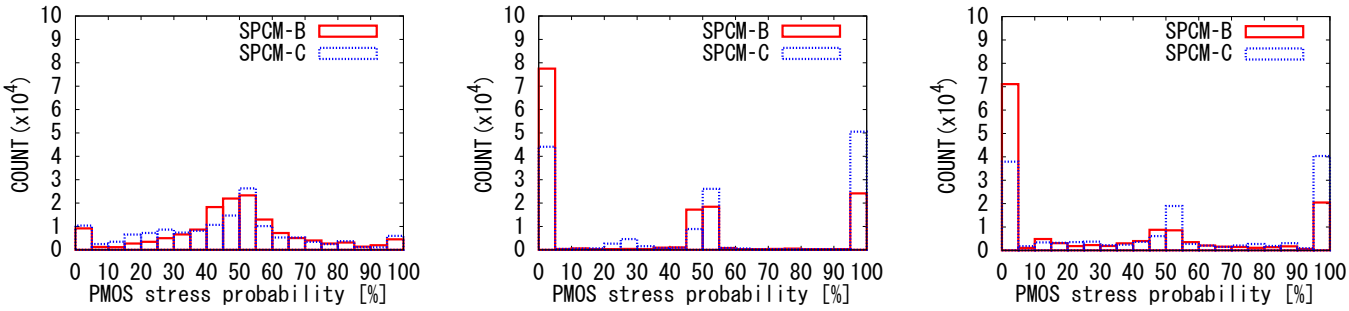
RESULTS OF STRESS PROBABILITY ESTIMATION IN FULL ADDER.

PMOS	Stress Probability [%]		
	SPCM-A	SPCM-B	SPCM-C
P1	50.0	0.0	100.0
P2	50.0	0.0	100.0
P3	50.0	0.0	100.0
P4	50.0	0.0	100.0
P5	50.0	0.0	100.0
P6	50.0	0.0	0.0

Furthermore, SPCM-A did not estimate stress probability as 100%, which may result in an optimistic estimation.

B.2.b **Stress probability distribution for all PMOSs.** Stress probability distributions for all PMOSs of D-FPU and AES are illustrated in Fig. 7 and Fig. 8, respectively. Note that PMOSs which compose feedback not related to timing in DFFs are excluded. In Fig. 7 and Fig. 8, samples of 100% / 0% stress probability are included in the range of over 95% / below 5%.

In Fig. 7(a), stress probability distributions are roughly uni-



(a) general operation (b) biased operation (only addition) (c) biased data

Fig. 7. Stress probability distribution (D-FPU).

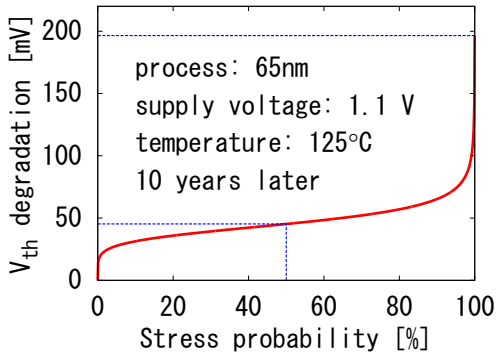


Fig. 6. V_{th} degradation versus stress probability of PMOS (65 nm, 1.1 V, 125°C, 10 years).

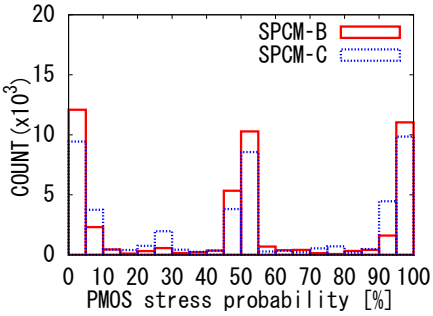


Fig. 8. Stress probability distribution (AES, general operation).

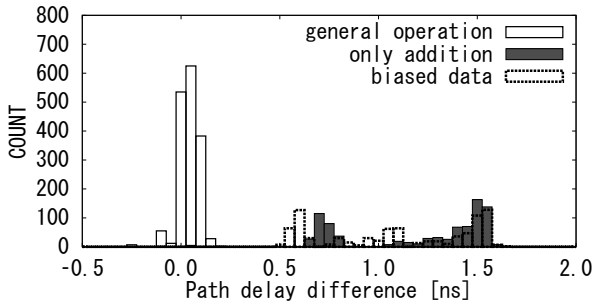


Fig. 9. Distribution of differences of path delay between SPCM-A and SPCM-C (D-FPU).

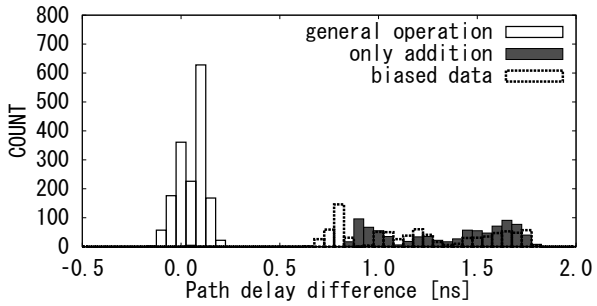


Fig. 10. Distribution of differences of path delay between SPCM-B and SPCM-C (D-FPU).

form, and there is a small number of PMOS transistors whose stress probability is 100%. In contrast, as shown in Fig. 7(b)(c), there are many PMOSs whose stress probability is either 0% or 100%. When operation or data is biased, the active circuit portion is limited, and a number of nets are fixed 0 or 1.

On the other hand, in Fig. 8, in spite of using random plain texts, there are large populations at 0% and 100%. It is supposed that the distributions of stress probability depend on circuit structures. Therefore, further evaluation of its dependency might be required as a future work.

B.2.c **Estimation accuracy of timing degradation.** Figure 9 gives a histogram of difference between path delays estimated by SPCM-A and -C, and Fig. 10 shows the difference between SPCM-B and -C in D-FPU. Here, top 2,000 paths in delay are selected for evaluation.

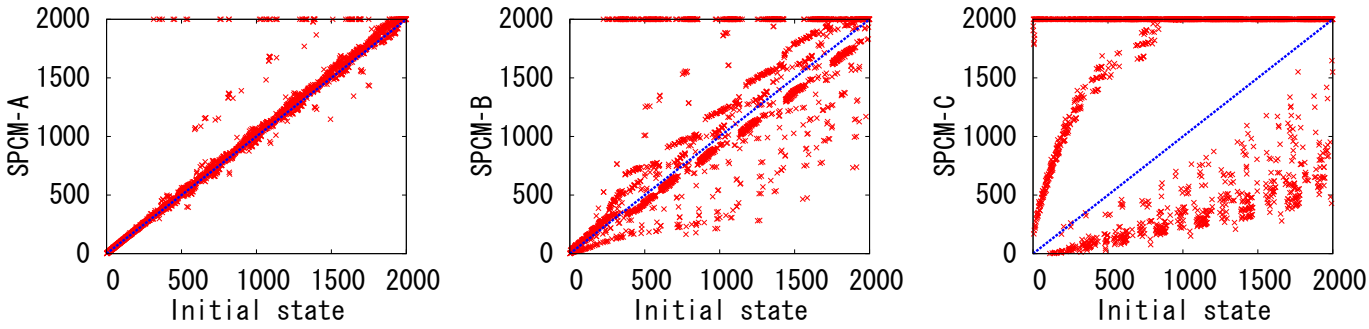
In Figs. 9 and 10, path delay differences both between SPCM-C and -A, and SPCM-C and -B are close to 0 when four arithmetic operations are done randomly. In case that all functions are performed, SPCM-A and -B are effective for delay prediction as shown in Table V. Meanwhile, when operation or operand is biased, the accuracies of SPCM-A and -B significantly degrade, and the error reaches 1.5 ns in this evaluation. We should note that instance-level stress probability consideration might induce unacceptable error in timing estimate, even though actual input patterns are used for evaluation.

C. Path delay order

Variation of path delay order due to NBTI is evaluated. The longest 2,000 paths in fresh D-FPU circuit are selected, and their rankings are compared with those after 10 years passed, supposing that D-FPU executes only addition at 125°C. Figure 11 illustrates the order of 2,000 paths, which correspond to about 2% of all paths. In the initial state, the difference of path delay between the 1st and 2000th is 0.950 ns. Each point represents each path. Note that the path whose ranking becomes over 2,000th after 10 years is plotted as a 2,000th path.

As shown in Fig. 11(a), when SPCM-A is adopted, the variation of path order is estimated to be small. On the other hand, when SPCM-C is adopted, the variation of path order is significant in Fig. 11(c).

When a circuit is activated locally by biased operation, delay of the path included in the inactive circuit portion, where logical values are fixed, becomes larger. Therefore, the path delay order



(a) SPCM-A (b) SPCM-B (c) SPCM-C

Fig. 11. Difference of path delay orders between fresh and 10-year-old circuits.

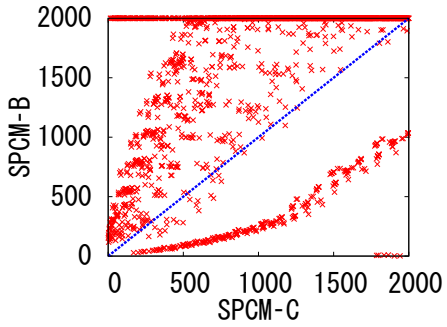


Fig. 12. Correlation between path delay orders estimated by SPCM-C and -B.

changes considerably. On the other hand, in SPCM-A, because stress probabilities of all PMOSs are set to 50%, NBTI degradation increases equally in all paths, and then the variation of path order is small as shown in Fig. 11(a). In contrast, Figs. 11(b) and 11(c) show a large disturbance of path order.

Finally, Fig. 12 plots the difference of path delay order between SPCM-B and -C after 10 years. The orders of SPCM-B and -C are less correlated. This result means that even if we fix timing-violated paths predicted by SPCM-B, different paths likely violate timing.

V. Conclusion

This paper evaluated how much stress probability consideration in NBTI delay degrading estimation impacts the accuracy, focusing on analysis granularity. Stress probability calculation and timing analysis were performed for two circuits at three granularity levels: circuit-level, instance-level, and transistor-level. Evaluation results showed that considerable errors may arise even though instance-level analysis is performed. For obtaining accurate timing degradation due to NBTI, transistor-level analysis both in stress probability computation and instance delay calculation is necessary. We also demonstrated that distribution of stress probabilities heavily depends operation of circuits.

VI. Acknowledgement

The authors would like to thank the project members of JST CREST of Kyoto University, Kyoto Institute of Technology, Nara Institute of Science and Technology, and ASTEM RI for their discussions. This study was partly supported by NEDO.

References

[1] S. Bhardwaj, W. Wang, R. Vattikonda, Y. Cao, and S. Vrudhula, "Predictive modeling of the NBTI effect for reliable design," *in Proc. CICC*, pp. 1047 – 1052, Aug. 2006.

[2] W. Wang, S. Yang, S. Bhardwaj, R. Vattikonda, S. Vrudhula, F. Liu, and Y. Cao, "The impact of NBTI on the performance of combinational and sequential circuits," *in Proc. ASP-DAC*, pp. 364 – 369, Jan. 2007.

[3] B. Zhang and M. Orshansky, "Modeling of NBTI-induced PMOS degradation under arbitrary dynamic temperature variation," *in Proc. ISQED*, pp. 748 – 779, Mar. 2008.

[4] A. Stempkovsky, A. Glebov, and S. Gavrilov, "Calculation of stress probability for NBTI-aware timing analysis ," *in Proc. ISQED*, pp. 714 – 718, Mar. 2009.

[5] W. Wang, S. Yang, and Y. Cao, "Node criticality computation for circuit timing analysis and optimization under NBTI effect," *in Proc. ISQED*, pp. 763 – 768, Mar. 2008.

[6] M. A. Alam and S. Mahapatra, "A comprehensive model of PMOS NBTI degradation," Microelectronics Reliability, vol. 45, pp. 71 – 81, 2005.

[7] J. H. Lee, W. H. Wu, A. E. Islam, M. A. Alam, and A. S. Oates, "Separation method of hole trapping and interface trap generation and their roles in NBTI reaction-diffusion model," *in Proc. IRPS*, pp. 745 – 746, 2008.

[8] S. Mahapatra, M. A. Alam, P. Bharath, T. R. Dalei, and D. Saha, "Mechanism of negative bias temperature instability in CMOS devices: degradation, recovery and impact of nitrogen," *in Proc. IEDM*, pp. 105 – 108, Dec. 2004.

[9] A. T. Krishnan, C. Chansellor, S. Chakravarthi, P. E. Nicollian, V. Reddy, A. Varghese, R. B. Khamankar, and S. Krishnan, "Material dependence of hydrogen diffusion: implications for NBTI degradation," *in Proc. IEDM*, pp. 688 – 691, Dec. 2005.

[10] W. Wang, V. Reddy, A. T. Krishnan, R. Vattikonda, S. Krishnan, and Y. Cao, "Compact modeling and simulation of circuit reliability for 65nm CMOS technology," IEEE Trans. Device and Material Reliability, vol. 7, pp. 509 – 517, Dec. 2007.

[11] Y. Cao, " Reliability mechanisms and the impact on IC designs," *in Proc. ASP-DAC*, Tutorial 4, pp. 342 – 372, Jan. 2009.

[12] W. Wang, Z. Wei, S. Yang, and Y. Cao, " An efficient method to identify critical gates under circuit aging," *in Proc. CICC*, pp. 735 – 740, Sept. 2007.

[13] K. Shinkai, M. Hashimoto, A. Kurokawa, and T. Onoye, "A gate delay model focusing on current fluctuation over wide-range of process and environmental variability," *in Proc. ICCAD*, pp. 47 – 53, Nov. 2006.

[14] Synopsys, Inc. : PathMill and PathMill Plus User Guide, Sept. 2005.

[15] Open Cores, "http://www.opencores.org/"