A Segmentation Kernel Fitting Technique to Circumvent Extreme Deviation from Exponentially Descent Tail Distribution

Worawit Somha
King Mongkut's Institute of Technology, Ladkrabang, Bangkok, Thailand
Email: worawit.so@kmitl.ac.th

Hiroyuki Yamauchi
Fukuoka Institute of Technology, Wajiro-Higashi, Higashi-ku, Fukuoka, Japan
Email: yamauchi@fit.ac.jp

***Abstract*—A segmentation kernel fitting technique has been proposed to circumvent an extreme deviation from the exponentially steeping descent tail distribution in the deconvolution. The proposed technique regenerates each segmented distribution line by finding the minimum of unconstrained multivariable function using derivative-free method. We decomposed the convolution effects of the two types of the minimum operating voltage variations caused by the spatially random threshold variation (VDD_{SPAT}) and the temporally random threshold variation (VDD_{TIME}), respectively. We discussed the VDD_{SPAT} and VDD_{TIME} effects on the SRAM fail-bit count (FBC) based on the decomposing results. It is found that the FBC estimation error for the proposed one can be reduced to almost 14-orders of magnitude smaller than that for the off-the-shell functions.**

***Index Terms*—deconvolution, Random telegraph noise, MATLAB-deconvolution function, SRAM margin variation**

I. Introduction

Ordinary SRAM fail-bit count estimation relied on the normal distribution model for simplicity. Thus, the convolution and the deconvolution could be simply calculated by the addition subtraction.

However, once the tail length of the *t* becomes no longer ignored in comparison with the *s*, the ordinary brief calculation method cannot be used anymore, as shown in Fig. 1 [1]-[6]. This is because the shape of the tail distribution of *t* doesn't follows the normal distribution but obeys log-normal typed distributions [1]-[3], [5], [6].

In this paper, VDD denotes for the SRAM operating voltage. *s* and *t* represent for the SRAM minimum VDD distributions VDD_{SPAT} and VDD_{TIME} effected by the spatially random threshold variation and the temporally random threshold variation, respectively.

To find the most accurate way instead of the ordinary brief estimation means, we experimentally measured the error of the deconvolution results for all available MATLAB functions [7]-[13]. Then, we investigated the two error dependencies of 1) tail length of VDD_{TIME} distribution *t* and 2) deconvolution algorithm to figure out the root causes for the error [14]-[19]. Based on the root causes analyses, we proposed a practical technique to circumvent the root causes for the errors. This paper is an extended version of work published in [20]. In this paper, VDD_{TIME} and the VDD_{SPAT} distributions are denoted by *t* and *s*. They are assumed to obey Gamma $G(\alpha, \beta)$ and Normal $\boldsymbol{N}(\sigma, \mu)$ distributions, respectively.

To normalize the variation amplitude of *t* to *s*, *s* is plot with distribution of $N(\sigma{=}1.0, \mu{=}0)$.

Three tail-length cases of *t* for VDD_{TIME}-1, VDD_{TIME}-2, and VDD_{TIME}-3 are assumed. They correspond to the variation amplitude distributions for the scaling MOSFET device dimension of a 40 nm, 15 nm, and 8 nm, respectively (see Fig. 1 (a), (b), and (c)).

The rest of the sections are divided into 4. Section 2 discusses the deconvolution error comparisons among all available off-the-shell functions. Section 3 proposes the deconvolution technique algorithm to circumvent the extreme error. The advantages over the conventional ones are discussed in Section 4. Section 5 concludes this paper.

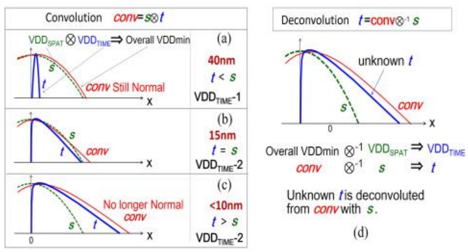


Figure 1. Convolution of VDD_{TIME} *t* with VDD_{SPAT} *s* i.e., $conv = s \otimes t$ in case of (a) $t<s$, (b) $t=s$, (c) $t>s$ and (d) deconvolution of *t* of *conv* with *s*, i.e., $t = conv \otimes^{-1} s$.

II. Off-the-Shell Deconvolution Functions

A. Algorithm Dependencies of the Errorures

We compared the algorithms for the 4 available functions: "deconv", "deconvreg", "deconvrls" and

Manuscript received March 11, 2018; revised Juan 21, 2018.

"deconvlucy". In this paper, we named these MATLAB functions as "off-the-shell" functions.

1) "deconv" uses an algebraic inverse way. This relies on the successive derivative operations. It causes a long polynomial division by zero and successive positive feedback error accumulation. It can be most easily seen the V-shaped error phenomena (see Fig. 2 (a) and Fig. 3 (a)).
2) "deconvreg" uses a regularized digital filter (see Fig. 2 (b)) to constrain the least square optimization error amount in the positive feedback loop between the estimated $t'(x)$ and the true one $t(x)$. This can limit the amplitude of the ringing phenomena but the level is not sufficient.
3) "deconvrls" (see Fig. 2 (c)) uniquely introduces the recursive adaptive filtering to reduce the noise amplification. This improves the error level but still is too large.
4) "deconvlucy" (see Fig. 2 (d)) uses the Richardson-Lucy (R-L) algorithm [13]-[17], [21]. Only this algorithm is categorized into the forward base deconvolution in comparison with other inverse problem based algorithm. However, this algorithm is not derivative free but still need the derivative operation for the maximum-likelihood (MLE) gradient iteration processes. Thus, this still causes an extreme deviation from the true value.

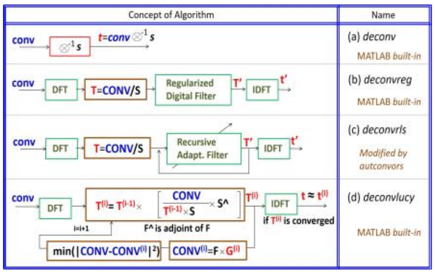
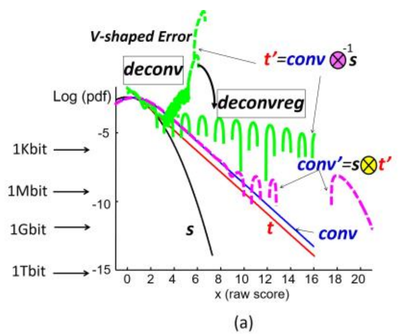
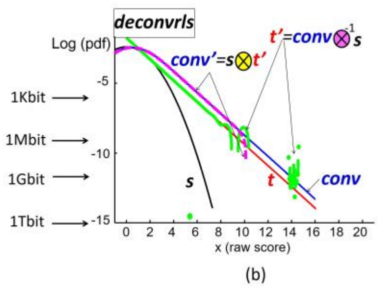


Figure 2. Off-the-shell functions: (a) deconv, (b) deconvreg, (c) deconvrls Modified function, and (d) deconvlucy.





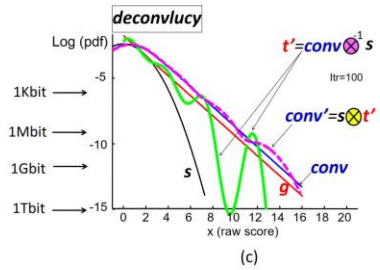


Figure 3. Deconvolution of t' with s and $conv$ with different algorithms: (a) deconv and deconvreg (b) deconvrls and (c) deconvlucy.

Fig. 3 plots the five-curves for s, t, $conv$, t', and $conv'$. Where, $t'=conv\otimes^{-1}s$ and $conv'=s\otimes t'$, respectively. The x-axis is for the "raw score" that represents the scale of the tail length of the t, $conv$, t', and $conv'$ in comparison with the normalized $s=N(\sigma=1.0,\ \mu=0)$. Y-axis is for the log-scaled probability density function (pdf). The tail length of t for the VDD_{TIME}-3 is longer than that for s for the VDD_{SPAT}, as can be seen in Fig. 3. The Y-positions indicated by the arrows for10^3 bit, 10^6 bit, 10^9 bit, and 10^{12} bit corresponds to the probability where only one bit fails of the whole bits, respectively.

It is found in Fig. 3 (a) that 1) "deconv" causes V-shaped error on t' in x>2, 2) "deconvreg" avoids the V-shaped error but still exhibits a ringing behavior, 3) the effects of the ringing error on the $conv'$ are significantly large in x>9. Since the curve of $conv'$ is deviated from the true line, the cumulative density function (cdf) of $conv'$ cannot be calculated. This means that the fail-bit count (FBC) cannot be precisely estimated.

It is found in Fig. 3 (b) that "deconvrls" reduces the ringing on t' line but still huge error remains in x> 9. As a result, the curve of $conv'$ is largely deviated from the true line.

It is also shown in Fig. 3 (c) that "deconvlucy" avoids the high-frequency noises unlike others. However, a low-frequency ringing error on t' and $conv'$ still remains. As a result, the cdf of $conv'$ that corresponds to FBC has large error.

From the requirement points of view for "SRAM FBC analyses", the pdf error level must be smaller than 10^{-12}. This is because a 10^{-12} corresponds to one-bit fail-probability for 1T-bit SRAM.

B. Discussions on Off-the-Shell Deconvolution Function

From the requirement points of view for "SRAM FBC analyses", the pdf error level must be smaller than 10^{-12}. This is because a 10^{-12} corresponds to one-bit fail-probability for 1T-bit SRAM. In that sense, it is concluded that the off-the-shell functions cannot be used for the FBC estimation of deep nanometer scaled SRAMs. This is because the tail length of the VDD_{TIME} distribution t becomes longer as the SRAM device size is scaled down.

Thus, to solve this problem, we propose a new deconvolution algorithm free from the derivative operation and the maximum-likelihood steep gradient iterations.

III. Forward Base Deconvolution Algorithm

The proposed algorithm eliminates the need of the differentiation, division, and the MLE steep gradient iteration, as shown in Fig. 4 (a).

1) $VDD_{TIME}^{(i)}$ is approximated by the gamma distribution, i.e., $VDD_{TIME}^{(i)}$=e×gamma[α, β] with three parameters of α (shape), β (inverse scale) and *e* (peak value).
2) Optimization problem to seek the parameter set FP[*e*, α, β] of fitting function $VDD_{TIME}^{(i)}$ for minimizing (|*conv-conv*$^{(i)}$|). Where $VDD_{TIME}^{(i)}$=*e*×gamma[α, β] and *conv*$^{(i)}$ is the convolution of $VDD_{TIME}^{(i)}$ with VDD_{SPAT}, i.e., (*conv'*=*s*⊗*t'*)
3) "fmin-search" in MATLAB to find the minimum of unconstrained multivariable function allowing a derivative-free method (|*conv-conv*$^{(i)}$|).

We named this "Fmin-search forward problem based deconvolution (FsrchDCV)" algorithm [18]-[20].

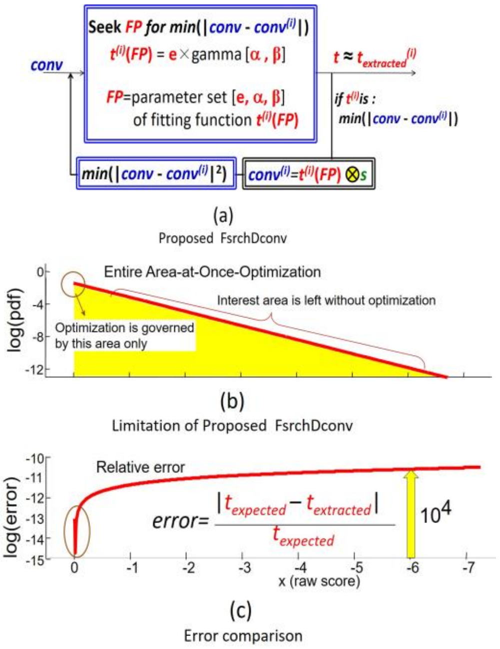


Figure 4. (a) Concept of fmin-search deconvolution algorithm (FsrchDCV), (b) VDD_{TIME}-2 deconvolution result using FsrchDCV, and (c) *x*-poistion dependency of Relative errors of the deconvolution error.

The VDD_{TIME}-2 deconvolution result is shown in Fig. 4 (b). The deconvoluted curves of the VDD_{TIME}-2 are smoothed without any ringing errors across the full range of *x*.

However, more probability density populated zone around *x*=0 accounts for a large fraction of the overall cdf. As a result, the pdf error is best reduced only around *x*=0. Unfortunately the errors in the tail region (*x*=6) is being left as it is. We cannot ignore the degree of error in the tail region. This is because the required level for the SRAM fail probability analysis is extremely low, e.g., pdf<10^{-12}.

In order to meet the requirements, a segmented FsrchDCV (SFsrchDCV) is newly introduced.

IV. Proposed Segmented FsrchDCV (SFsrchDCV) Algorithm

The proposed algorithm steps consist of the following four steps [20]:

1) Fitting curve for $VDD_{TIME}^{(i)}$ is divided into N segmentation ($VDD_{TIME\ k}^{(i)}$: *k*=1 to *N*).
2) The last line-segment of ($VDD_{TIME\ 1}^{(i)}$: $VDD_{TIME\ k\text{-}1}^{(i)}$) is extended with the new segmentation of $VDD_{TIME\ k}^{(i)}$.
3) The line of *conv*$^{(i)}$=[seg($VDD_{TIME\ 1}^{(i)}$: $VDD_{TIME\ k\text{-}1}^{(i)}$) +$VDD_{TIME\ k}^{(i)}$] ⊗ VDD_{SPAT} (see Fig. 5).
4) The process of finding the best $VDD_{TIME}^{(i)}$ takes the step by step manner from *k*=1 to *N*.

Once the best $VDD_{TIME\ k}^{(i)}$ is found in each segment, its value is temporally fixed when seeking the next $VDD_{TIME\ k+1}^{(i)}$.

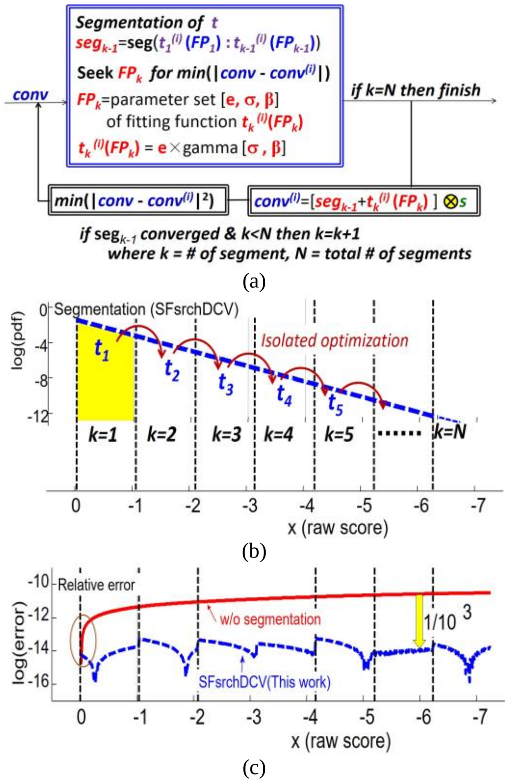


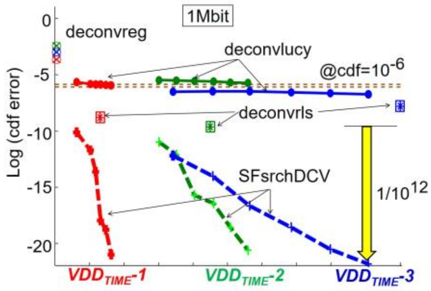
Figure 5. (a) Flow, (b) Concept of proposed segmented FsrchDCV algorithm (SFsrchDCV), and (c) deconvolution relative errors between FsrchDCV and SFsrchDCV (this work).

Each optimization step is isolated from the populated zone. Thus the optimization process can be well focused even at the end of the lines (*k*=*N*) in the interest zone.

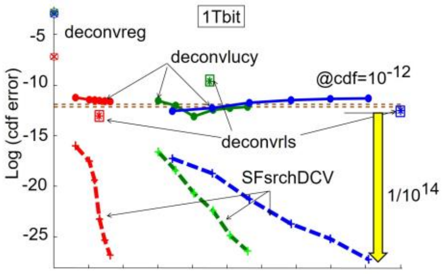
The 3-orders of magnitude error reduction is achieved in comparison with the non-segmentation case, as shown in Fig. 5 (c). This results from that each optimization can be focused in individual segment unlike the case without segmentation manner. As a result, the error level has no *x* position dependency.

V. CDF Errors and Iterations for VDD_{TIME}-1,2,3

The convergence speed of the cdf error reduction are compared among the different algorithm. The tail length dependencies are compared among the VDD_{TIME}-1, VDD_{TIME}-2, and VDD_{TIME}-3.



(a)



(b)

Figure 6. Cdf-errors comparisons among the proposed SFsrchDCV, "deconvreg", "deconvrls", and "deconvlucy" for VDD_{TIME}-1, VDD_{TIME}-2, and VDD_{TIME}-3: (a) 1K-bit × 1000 pieces=1Mbit and (b) 1Gbit × 1000 pieces=1Tbit.

The memory bit density dependencies is also compared between 1Mbit (1000-pieces of 1K-bit SRAM) and 1T-bit (1G-bit SRAMs), as shown in Figs. 6a-6b. It is shown that the proposed SFsrchDCV can reduce the cdf error by 10^{12} to 10^{14}-fold in comparison with off-the-shell functions. It is found that the errors can be reduced with increasing the iteration cycles by using the proposed scheme. This results from the reduced optimization width allowing an increase in resolution of the deconvolution step. This is the clear contribution of the proposed algorithm.

VI. Conclusion

The segmented deconvolution technique (SFsrchDCV) is proposed which is free from any derivative operations causing the ringing error. The effectiveness of the SFsrchDCV algorithm is shown based on the error comparison results. It is found that the proposed SFsrchDCV provides a 3-orders and 14-orders of magnitude error reduction in the VDD_{TIME} deconvolution in comparison with FsrchDCV and the off-the-shell functions, respectively.

Acknowledgment

The authors wish to thank the technological staffs and students in both of Worawit Laboratory in KMITL and Yamauchi Laboratory in Fukuoka Institute of Technology.

References

[1] V. M. Santen, *et al.*, "Reliability in super- and near-threshold computing: A unified model of RTN, BTI, and PV," *IEEE Trans. on Circuits and Systems I: Regular Papers*, vol. 65, no. 1, pp. 293-306, 2018.

[2] Q. Tang, *et al.*, "Characterizing the impact of RTN on logic and SRAM operation using a dual ring oscillator array circuit," *IEEE Journal of Solid-State Circuits*, vol. 52, no. 6, pp. 1655-1663, 2017.

[3] P. Weckx, *et al.*, "Relaxation of time-dependent NBTI variability and separation from RTN," in *Proc. Digest of IEEE Int. Reliability Physics Symp.*, 2017, pp. XT9.1-XT9.5.

[4] K. Takeuchi, "Comprehensive SRAM design methodology for RTN reliability," in *Proc. Digest of IEEE Symp. on VLSI Technology*, 2011, pp. 130-131.

[5] X. Wang, "RTS amplitude distribution in 20nm SOI FinFETs subject to statistical variability," in *Proc. SISPAD* 2012, pp. 296-299.

[6] X. Wang, "Simulation study of dominant statistical variability sources in 32-nm high-k/metal gate CMOS," *IEEE Electron Device Letters*, vol. 33, no. 5, pp. 643-645, 2012.

[7] J. Zhang, *et al.*, "Learning fully convolutional networks for iterative non-blind deconvolution," in *Proc. Digest of IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 6969-6979.

[8] Q. Shi, *et al.*, "Application of iterative deconvolution for wire fault location via reflectometry," in *Proc. Digest of IEEE Int. Symp. on Instrumentation & Measurement, Sensor Network and Automation (IMSNA)*, 2012, pp. 102-106.

[9] H. Yu, *et al.*, "A comprehensive study of blind and non-blind deconvolution methods applied to spectrum extraction," in *Proc. Digest of 13th Int. Conf. on Computational Intelligence and Security (CIS)*, 2017, pp. 484-487.

[10] Y. Hang, *et al.*, "The control regularization-term iterative algorithm for image deconvolution" in *Proc. Digest of 4th Int. Conf. on Systems and Informatics (ICSAI)*, 2017, pp. 1191-1196.

[11] W. Somha and H. Yamauchi, "Convolution/deconvolution SRAM analyses for complex Gamma mixtures RTN distributions," in *Proc. of Digest of ICICDT-2013*, 2013, pp. 33-36.

[12] W. Somha and H. Yamauchi, "A V-shaped deconvolution error suppression technique for SRAM analyses of long-tail Gamma mixtures random telegraph noise distribution effects," in *Proc. Digest of ITC-CSCC-2013*, 2013, pp. 297-299.

[13] D. A. Fish, *et al.*, "Blind deconvolution by means of the Richardson–Lucy algorithm," *Journal of the Optical Society of* America *A*, vol. 12, no. 1, pp. 58–65, 2009.

[14] A. Mosleh, *et al.*, "Explicit ringing removal in image deblurring," *IEEE Trans. on Image Processing*, vol. 27, no. 2, pp. 580-593, 2018.

[15] K. Somasundaram, *et al.*, "Brain portion segmentation from magnetic resonance images (MRI) of human head scan using Richardson Lucy deconvolution and intensity thresholding," in *Proc. Digest of Int. Computer Science and Engineering Conf. (ICSEC)*, 2016, pp. 1-5.

[16] R. L. White, "Image restoration using the damped Richardson-Lucy method," *The Restoration of HST Images and Spectra II, Space Telescope Science Institute*, 1994, pp. 104-110.

[17] F. Dell'Acqua1, "A modified damped Richardson-Lucy algorithm to improve the estimation of fiber orientations in spherical deconvolution," *Proc. Intl. Soc. Mag. Reson. Med.*, vol. 16, pp. 1-10, 2008

[18] A. Tselousov, *et al.*, "Kernel estimate for image restoration using blind deconvolution," in *Proc. Digest of IEEE Conf. of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus)*, 2017, pp. 753-757.

[19] W. Somha, H. Yamauchi, and Y. Ma, "Adaptively segmented forward problem based non-blind deconvolution technique for analyzing SRAM margin variation effects," in *Proc. Digest of Int. Soc. Design Conf.*, November, 2013, pp. 184-187.

[20] H. Yamauchi and W. Somha, "Comparative study on deconvolution function dependencies of RTN/RDF effect estimation errors in analyzing sub-nm-scaled SRAM margins," in *Proc. IEEE 57th Int. Midwest Symp. on Circuits and Systems (MWSCAS)* 2014, pp. 230-233.

[21] H. Yamauchi and W. Somha, "A technique to solve issue of Richardson-Lucy deconvolution for analyzing RTN effects on SRAM margin variation," in *Proc. 5th IEEE Latin American Symp. on Circuits and Systems*, Feb. 2014, pp. 1-6.

Worawit Somha received the Ph.D. degree in engineering from Fukuoka Institute of Technology, Fukuoka, Japan, in 2015 and the master degree in electrical engineering from King Mongkut's Institute of Technology Ladkrabang (KMITL), Bangkok, Thailand. His master thesis was on "Vector Quantizers for Speech Coding and there Implementation on TMS-320C30". Since 1995 he has given a lecture for bachelor degree student in subject of "Introduction to Digital Signal Processing" at KMITL as an assistance professor, and his research area is speech coding. Since 1997 he has worked with the company in the position of consulting engineering.

Hiroyuki Yamauchi received the Ph.D. degree in engineering from Kyushu University, Fukuoka, Japan, in 1997. His doctoral dissertation was on "Low Power Technologies for Battery-Operated Semiconductor Random Access Memories". In 1985 he joined the Semiconductor Research Center, Panasonic, Osaka, Japan. From 1985 to 1987 he had worked on the research of the submicron MOS FET model-parameter extraction for the circuit simulation and the research of the sensitivity of the scaled sense amplifier for ultrahigh-density DRAM's which was presented at the 1989 Symposium on VLSI Circuits. From 1988 to 1994, he was engaged in research and development of 16-Mb CMOS DRAM's including the battery-operated high-speed 16 Mbit CMOS DRAM and the ultra low-power, three times longer, self-refresh DRAM which were presented at the 1993 and 1995 ISSCC, respectively. He also presented the charge-recycling bus architecture and low-voltage operated high-speed VLSI's, including 0.5V/100MHz-operated SRAM and Gate-Over-Driving CMOS architecture, which were presented at the Symposium on VLSI Circuits in 1994 and 1996, respectively, and at the 1997 ISSCC as well. After experienced general manager for development of various embedded memories, eSRAM, eDRAM, eFlash, eFeRAM, and eReRAM for system LSI in Panasonic, he has moved to Fukuoka Institute of Technology and become a professor since 2005. His current interests are focused on study for variation tolerant memory circuit designs for nano-meter era. He holds 212 Patents including 87 U.S. Patents and has presented over 70 journal papers and proceedings of international conferences including 10 for ISSCC and 11 for Symposium on VLSI Circuits. Dr. Yamauchi received the 1996 Remarkable Invention Award from Science and Technology Agency of Japanese government and the highest ISOCC2008 Best Paper Award.

He had been serving a program committee of ISSCC for long periods, from 2002 through 2009.

He served a program committee of IEEE Symposium on VLSI Circuits from 1998 through 2000 and has come back and been serving again since 2008. He is also serving A-SSCC since 2008.