Analysis and Mitigation of Process Variation Impacts on Power-Attack Tolerance

Lang Lin
Department of Electrical and Computer Engineering,
University of Massachusetts Amherst, MA
llin@ecs.umass.edu

Wayne Burleson
Department of Electrical and Computer Engineering,
University of Massachusetts Amherst, MA
burleson@ecs.umass.edu

ABSTRACT
Embedded cryptosystems show increased vulnerabilities to implementation attacks such as power analysis. CMOS technology trends are causing increased process variations which impact the data-dependent power of deep submicron cryptosystem designs. In this paper, we use Monte Carlo methods in SPICE circuit simulations to analyze the statistical properties of the data-dependent power with predictive 45nm CMOS device and ITRS process variation models. In addition to the "measurement to disclosure" (MTD) used in [3], we define a lower level metric, Power-Attack Tolerance (PAT), to model both dynamic power and leakage power data-dependence. We show that the PAT of a typical cryptographic component implementation using CMOS standard-cells can significantly deteriorate due to process variations, thus increasing the component's vulnerability to power attacks. Power-attack-resistant logic styles (e.g. SABL [9]) have been developed which increase PAT by an order of magnitude by balancing power consumption at the gate level with considerable overhead. However in the presence of process variations, the degradation probability of MTD is 57%. To mitigate this problem, we demonstrate a transistor sizing optimization method that can reduce such negative impacts to only 18% with minimal power and area overhead.

Categories and Subject Descriptors
B.6 [**Hardware**]: Logic Design; B.7 [**Hardware**]: Integrated Circuits; E.3 [**Data**]: Data encryption.

General Terms
Design, Security, Verification.

Keywords
Process variation, Monte Carlo simulation, differential power analysis, transistor sizing.

1. INTRODUCTION
Embedded system security is realized by an embedded cryptosystem to execute the encryption and decryption computations with a secret key. By sharing the key with only trusted parties, security can be guaranteed theoretically. However, since every computation process is inevitably accompanied with an amount of power consumption, the logic values involved in the computations can be reflected by specific power patterns. This physical phenomenon enables a typical implementation attack, called "power analysis attack" [1], to extract the secret key by exploiting the correlation between the logic values of the key bits and the measured run-time power traces. The effectiveness of power analysis attacks is determined by two parameters of a cryptosystem implementation [2]:

1). Signal-to-noise ratio (SNR), which is the ratio of the data-dependent power to the data-independent power. For a cryptosystem embedded in an SoC, the on-die power regulators and many non-crypto integrated circuits can lead to large amount of the total data-independent power, which will prevent a simple power analysis attack. To deal with this, differential power analysis (DPA) [1] attack is introduced to statistically collect the subtle data-dependent power and filter out the data-independent power. DPA attacks can efficiently extract the secret key by non-invasive methods and off-the-shelf tools. The attack starts from making a guess of the secret key, and use this key guess to predict the values of some bits (known as the "selection function") involved in the crypto computations. The prediction is used to group the measured power traces into data-dependent and data-independent categories. By accumulating the data-dependent power and neutralizing the data-independent power, a differential power curve (DPC) is generated with respect to a key guess. If the key guess is correct, the corresponding DPC is distinguishable from other DPCs. If no distinguishable DPC can be found, more power traces need to be analyzed to overcome the low SNR. The number of power traces needed to extract the key is named the "measurement to disclosure" (MTD) in [3], which is inversely proportional to the SNR [4].

2). The time t_c, during which the data-dependent power appears on each measured power trace. DPA attacks make use of transient power, instead of time-average power, to extract the secret key. Thus, disarrangement of t_c on each power trace can prevent power analysis. For example, if a random clock-cycle-based delay is inserted to disturb the synchronization of the power traces, DPA attacks based on data-dependent dynamic power become difficult. However, DPA attacks based on data-dependent active leakage power [5] are hardly affected by random delay insertion. The active leakage power can be measured between a clock rising edge and a neighboring clock falling edge, when all the logic gate switching behaviors in the current clock cycle have settled down

DAC'09, July 26–31, 2009, San Francisco, California, USA.

before the next switching behavior occurs. The attackers can easily defeat the random delay approach, since the duration of active leakage power is always much longer than the introduced delay uncertainty.

In deep submicron technologies, process variations become a critical concern and most deterministic design features become probabilistic. To model the process variations, the design features are usually assigned with a normal distribution and the probability design functions (PDF) of some circuit performance metrics (such as power and delay) can be derived with Monte Carlo simulations. Although Monte Carlo simulation is time-consuming, it can provide the best theoretical accuracy by modeling both systematic and random process variations.

Process variations can significantly impact both dynamic and active leakage power distributions of fabricated ICs [6]. Intra-die process variations impact every transistor on a single die differently, while inter-die process variations impact all transistors on one die in the same way. In the context of DPA attacks, process variations can also impact the data-dependence of both dynamic and leakage power. Such impacts are not reflected by the measurement noise or SNR of the same chip, but they can lead to a spectrum of SNR across different chips. Process variations have already been predicted to influence the power model during the security evaluation of embedded cryptosystem design [7]. In [8], process variations are modeled as data-independent power that does not impact power attacks. However, in this paper, we use a statistical approach to demonstrate that process variations can increase the inherent SNR of standard-cell CMOS (sCMOS) gates to facilitate power attacks. More importantly, due to the negative impacts of process variations, many existent gate-level countermeasures against DPA attacks can fail. For example, sense-amplifier-based logic (SABL) gates [9] can resist DPA by balancing the power consumption of differential pairs with symmetrically sized transistors to minimize the inherent SNR. In [3] and [10], differential pair routing approaches for ASIC and FPGA are proposed to balance the interconnect capacitance to achieve even lower SNR. However, we demonstrate here that the presence of process variations is an additional factor to impair the power balance and thus can deteriorate the power-attack tolerance (PAT) of all kinds of power-attack-resistant logic styles. As far as we know, no prior work has considered process variation impacts during the early design phase of embedded cryptosystems.

The remainder of this paper is organized as follows. Section 2 defines and analyzes the statistical PAT. Section 3 shows two cases of simulation-based DPA attack to demonstrate the negative impacts of process variations on MTD. Section 4 proposes a transistor sizing method to compensate for the impacts, following by conclusions and suggestions for future works in Section 5.

2. PRELIMINARIES

The power model of sub-90nm CMOS devices considering both dynamic power and active leakage power is given as:

$$P_{total} = P_{dyn} + P_{leak} = f_{0\to1} \cdot C_L \cdot V_{dd}^2 + (I_{sub} + I_{gate} + I_{BTBT}) \cdot V_{dd} \quad (1)$$

Dynamic power is consumed when a 0-to-1 logic transition happens at a circuit node to charge the node capacitance. Active leakage power is consumed by each transistor throughout the circuit functioning time. Both power components are highly data-dependent. For a logic gate with N inputs, the data-dependent dynamic power has $d=2^{2N}$ values, if we consider all of the logic transitions. The data-dependent leakage power has $l=2^N$ values, if we consider all of the logic combinations. The nominal dynamic or leakage power is calculated by the mean of the data-dependent power values.

With the aggressive technology scaling, process variations can significantly impact the power distribution. The most dominant process variations on power distribution are the threshold voltage V_{th} and the effective channel length L_{eff} [11]. First-order design simulations always model them as a normal distribution. Then the dynamic power will meet a normal distribution, and the leakage power will meet a lognormal distribution [12]. As reported from the ITRS [13] and industry sources, the 3σ intra-die variation of V_{th} and L_{eff} can be as large as 42% and 12% in 45nm technology.

2.1 Statistical Power-Attack Tolerance

In the context of power analysis attacks, we are particularly interested in the data-dependence of power consumption instead of mean power. Strong data-dependent power implies large SNR that facilitates power attacks. For an N-input logic gate, the SNR of dynamic power and leakage power can be calculated by:

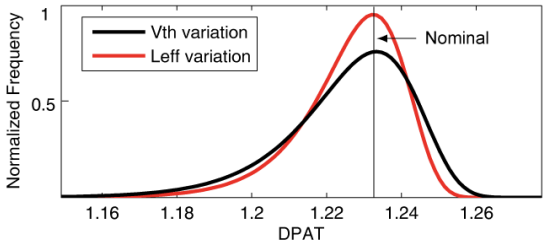
$$SNR_{dyn} = \frac{\sigma(P_d)}{\mu(P_d)}, SNR_{leak} = \frac{\sigma(P_l)}{\mu(P_l)}. \quad (d = 2^{2N}, l=2^N) \quad (2)$$

where $\sigma(P)$ and $\mu(P)$ are the standard deviation and mean of dynamic/leakage power, with respect to all logic transitions/combinations. We define a metric called power-attack tolerance (PAT) to reflect the ability of a circuit to tolerate power attacks. PAT equals to the inverse of SNR. We can respectively use DPAT and LPAT for dynamic power and leakage power. They can be calculated by:

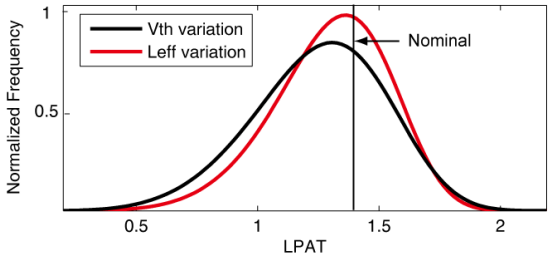
$$DPAT = \frac{\mu(P_d)}{\sigma(P_d)}, LPAT = \frac{\mu(P_l)}{\sigma(P_l)} \quad (3)$$

Each logic gate has the nominal DPAT and LPAT without process variations. To study the statistical PAT under V_{th} and L_{eff} variations, we perform 8000 iterations of Monte Carlo simulations on an NAND2 gate to achieve enough statistical accuracy. We use NAND2 gate because it can generally represent on-chip logic gates. The transistor parameters in our simulation are based on the Predictive Technology Model (PTM) [14]. In Figure 1, we show the PDFs of DPAT and LPAT. We find that the curves have different skews according to the mean value, which cannot be fitted to a normal or lognormal distribution function. The reason is that DPAT and LPAT do not have an analytical dependence on either V_{th} or L_{eff}. Therefore, we use the Weibull distribution function [15] for curve fitting. The PDF of a Weibull random variable x is expressed by:

$$f(x;\alpha,\beta) = \begin{cases} \frac{\alpha}{\beta} x^{\alpha-1} e^{-x^\alpha/\beta}, & x \geq 0 \\ 0, & x < 0 \end{cases}$$
$$\mu = \beta^{1/\alpha} \Gamma\left(\frac{\alpha+1}{\alpha}\right) \quad (4)$$
$$\sigma^2 = \beta^{2/\alpha} \left[\Gamma\left(\frac{\alpha+2}{\alpha}\right) - \Gamma^2\left(\frac{\alpha+1}{\alpha}\right) \right]$$



(a)



(b)

Figure 1. Process variation impacts on: (a) DPAT; (b) LPAT.

In the above expression, α is the shape parameter and β is the scale parameter. The flexibility of the two parameters makes the Weibull distribution a general-purpose function, which can be accommodated to normal distributions, exponential distributions and other distributions.

We show the statistical metrics of the resulting PAT distribution function in Table 1. We can use μ(PAT) to represent the average PAT under process variations, and σ(PAT) to represent the PAT uncertainty under process variations. We find that μ(LPAT) is negatively biased from the nominal LPAT, which indicates a degraded LPAT due to process variations. The reason behind this is that process variations have significant impacts on leakage power and such impacts vary for different input patterns. To further quantify the degradation of PAT, we calculate the cumulative probability that PAT is less than the nominal value. The degradation probability of DPAT and LPAT are as large as 55% and 66% respectively. This implies that process variations have a negative impact on the PAT of sCMOS gates.

Table 1. Statistical metrics of DPAT and LPAT of sCMOS

	DPAT	**LPAT**
	V_{th} / L_{eff} variation	V_{th} / L_{eff} variation
Nominal	1.23 / 1.23	1.38 / 1.38
μ (PAT)	1.23 / 1.23	1.25 / 1.30
σ (PAT)	0.017 / 0.014	0.277 / 0.243
P(degradation)	54% / 55%	66% / 60%

2.2 Impacts on SABL

SABL is one of the differential dynamic logic styles that have almost data-independent dynamic power. The schematic of a SABL AND-NAND gate is shown in Figure 2. SABL gates are refreshed during the pre-charge stage to de-correlate the power dependence on logic transition patterns. The pull-down network implements the differential logic that can be sensed by the latch

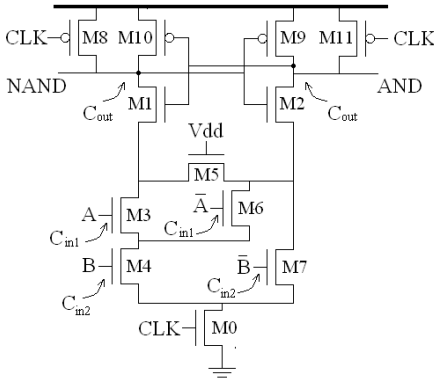
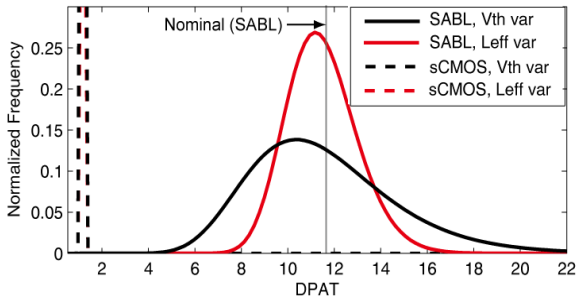


Figure 2. SABL AND-NAND gate without process variations.

on top. The load capacitance (e.g. C_{in1}, C_{in2} and C_{out}) of each differential pair is balanced by full-custom design process, so that the dynamic power for all input patterns is equal.

By introducing the same process variations into the power model, we simulate the PDFs of DPAT and LPAT for a SABL AND-NAND gate. The results are shown in Figure 3, together with the PDFs of the sCMOS NAND for comparison. The PDFs of sCMOS are partly shown, because they are too high compared with those of SABL. The nominal PAT of SABL is inherently 10x larger than that of sCMOS. However, the worst-case process corner can significantly reduce this advantage. Moreover, we can find that LPAT of SABL can deteriorate as low as sCMOS equivalent gate. This indicates that process variations have much more negative impacts on the data-dependent leakage power than the data-dependent dynamic power, especially for large circuits.



(a)

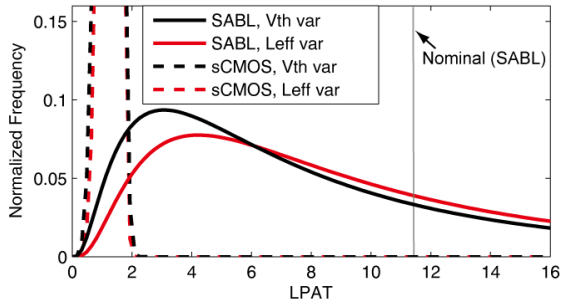


Figure 3. Process variation impacts on SABL gate compared with sCMOS gate: (a) DPAT; (b) LPAT.

We summarize the statistical metrics of DPAT and LPAT in Table 2. Comparing the value of σ, we find LPAT to have a much larger uncertainty due to process variations. The degradation

probability of SABL gates is as large as 59-71%, even more than sCMOS.

Table 2. Statistical metrics of DPAT and LPAT of SABL

	DPAT	LPAT
	V_{th} / L_{eff} variation	V_{th} / L_{eff} variation
Nominal	11.82 / 11.82	11.68 / 11.68
μ (PAT)	11.55 / 11.48	10.75 / 12.45
σ (PAT)	3.15 / 1.52	10.3 / 11.98
P(degradation)	59% / 61%	71% / 64%

2.3 Inter-Die Variation Impacts

Although DPAT of SABL gates is less vulnerable to intra-die variations than LPAT, it can further deteriorate with inter-die variations. Let us assume that every transistor in a SABL AND-NAND has 10% less Vth than the nominal value, in addition to the intra-die Vth variation used before. In Figure 4, we find that the PDFs with inter-die variations make a left shift to the worse region. In Table 3, we summarize the degradation of PAT due to both intra-die and inter-die variations. The average PAT and the uncertainty of PAT both deteriorate, with the only exception of the DPAT uncertainty. In all, the degradation probability is increased.

Apparently, PAT can be improved by adding positive inter-die Vth variation. This indicates that the process variation can be used as a "knob" to calibrate the PAT of an embedded system. Note that the increased Vth can lead to performance degradation and power reduction too. Therefore, the optimization of PAT must take other design trade-offs into account.

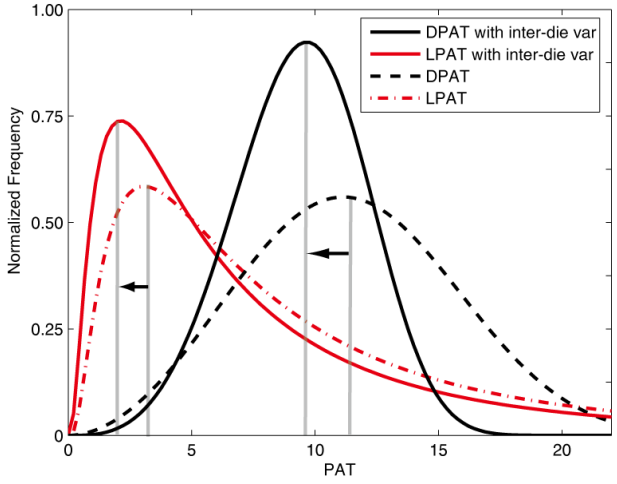


Figure 4. Impacts of additional inter-die process variation on SABL gate

Table 3. Statistical metrics of DPAT and LPAT

	DPAT	LPAT
Δμ(PAT)	-2.15	-1.76
Δσ(PAT)	-0.53	1.12
P(degradation)	70%	85%

3. CASE STUDIES

The process variation impacts on the gate-level PAT result in a statistical MTD of DPA attack on a cryptosystem implementation. However, since the impacts on each gate cell of a cryptosystem are not correlated, it is not easy to use the gate-level PAT to derive the statistical MTD. To study this statistical MTD, we need to simulate multiple sets of power traces of a cryptosystem by Monte Carlo method. Then we perform simulation-based DPA attacks on each set of power traces to acquire the corresponding MTD. As an example, we implement a standalone DES (Data Encryption Standard) cryptosystem using 45nm sCMOS library and perform a DPA attack on the fifth substitution-box (Sbox5) during the first round of DES. We use Synopsys Design Compiler to perform the logic synthesis and HSpice to simulate the transient power traces. Piece-Wise-Linear (PWL) voltage source elements are called in HSpice to generate random input plaintexts for the Sbox5 sub-circuit. Then Perl scripts can parse the power trace data and handle the DPA tasks, such as making key guesses, grouping power traces and generating DPCs. The selection function used in our simulations is the Hamming Weight of the Sbox5 ciphertext. The MTD is finally reported, if either a peak dynamic power or a maximum leakage power stands out from the DPCs to indicate a successful key extraction (we set the key as 12 in this example).

Without process variations, our simulation-based DPA attack on the sCMOS DES component results a MTD of 120. Note that the simulated MTD is much smaller than the MTD in a real DPA attack, because a real embedded system has much lower SNR due to the power consumption of non-crypto circuits. By introducing 42% Vth intra-die variation, we perform 30 Monte Carlo simulations to get the power traces. It takes 34 hours to finish all these simulations on an Intel 3GHz Pentium 4 processor with 3GB of memory. For each set of power traces, we find out the corresponding MTD. The distribution of the MTDs is shown in Figure 5. The worst-case MTD is as low as 88, which is 27% smaller than the nominal MTD.

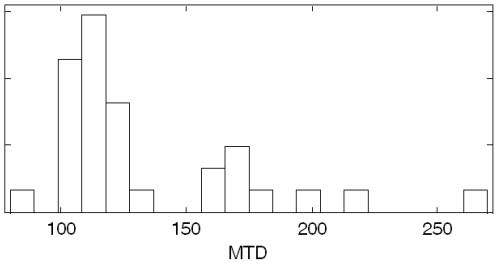


Figure 5. MTD distribution of sCMOS DES

To make a comparison, we design a SABL DES cryptosystem by replacing the sCMOS gates with SABL gate sub-circuits. The nominal MTD is 2300. With the same Vth variation, the MTD distribution is shown in Figure 6. The worst-case MTD is 1612, which deteriorates by 30%.

It is difficult to find a fitting curve for the resulting MTD distribution. A high confidence interval to estimate a statistical MTD cannot be achieved, because even thousands of Monte Carlo simulations on a large circuit can take forbidden amount of time. However, the central limit theorem ensures us that the estimation of the mean of MTD can be fairly accurate. Thus, we should use the degradation probability of P(MTD<μ(MTD)) to quantify our results. The degradation probability of MTD is 45% for the sCMOS DES and 57% for the SABL DES. It turns out that

process variations have more negative impacts on SABL DES. Note that the above results only consider the intra-die variation. We believe that the addition of inter-die variation can result even higher degradation probability of MTD. Although the process variation impacts cannot be effaced, they can be partially mitigated in secure embedded cryptosystem design.

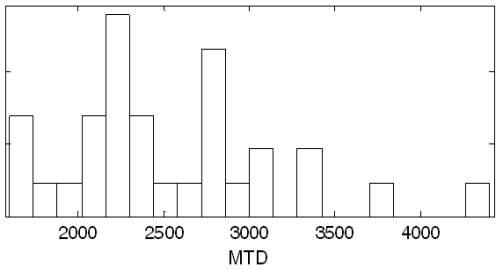


Figure 6. MTD distribution of SABL DES

4. MITIGATIONS

General design methods to deal with process variation impacts on deep submicron circuits are listed in [16]. The increase of transistor channel length can mitigate the L_{eff} variation. For short-channel transistors, the increase of channel length can also increase the threshold voltage and thus mitigate the V_{th} variation [17]. Therefore, we intend to use the transistor sizing method to mitigate process variation impacts on power-attack tolerance. To solve this transistor sizing optimization problem, we can use either the SPICE-based simulation approach or the statistical power estimation approach based on approximate analytical solutions [18]. The former is less efficient than the latter. However, we prefer the SPICE-based approach because power analysis attacks exploit key-dependent power from high-resolution power traces and the evaluation of the power-attack tolerance relies on time-accurate transient power profiles. To design a special-purpose embedded cryptosystem demanding high-level security, the optimization accuracy is more important than the optimization efficiency.

During the optimization procedure, the delay and implementation area penalties caused by sized-up transistors need to be considered. Therefore, we initially set a global transistor sizing constraint to be 10% of the minimum channel length (5nm for 45nm technology), with 1nm sizing resolution [19]. This global constraint can be relaxed if the performance and area are not critical design metrics.

4.1 Gate-level Mitigation

In this work, the ultimate goal is to apply a transistor-level sizing method to decrease the degradation probability of MTD caused by realistic process variations. We assume that each transistor in a logic gate can be individually sized, and the optimized sizing is then applied to the same gate in the design. To optimize the PAT of each gate, two design aims should be considered:

1. Decrease the uncertainty of PAT;
2. Increase the average PAT.

We want to achieve both these two aims, because the failure of one could compromise the success of the other. This can be illustrated in Figure 7. Assume that the original non-optimized PAT distribution of a certain type of logic gate is shown as Design 0. Design 1 is optimized for aim 1, but the degradation probability of PAT is more than 50%. Design 2 is optimized for aim 2, but a small number of chips can have extremely low PAT (less than 5). Therefore, we should optimize the metric $\mu(PAT)/\sigma(PAT)$ to achieve both the two aims at the gate level.

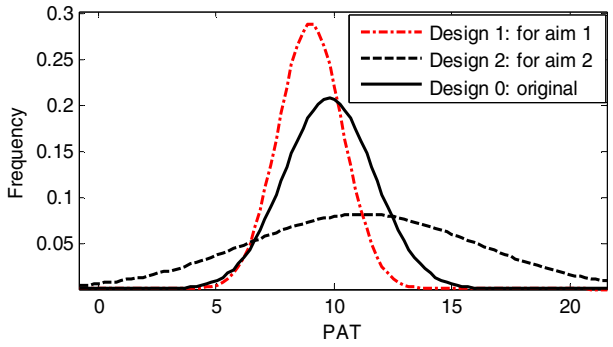


Figure 7. PAT optimization for different design aims.

We demonstrate a PAT optimization of a SABL DES cryptosystem as an example. As a power-attack-resistant logic, the inherent DPAT of SABL gates is ideally infinite (not infinite for a real chip with process variations). The inherent LPAT is not addressed in the original SABL design. As we show before, LPAT is much more sensitive to process variations than DPAT. Therefore, we emphasize more on optimizing the LPAT of SABL gates. To avoid the area overhead of sizing up all the transistors, we focus on the transistors directly connected to the input signals. Thus, we only optimize the length of transistors M3, M4, M6 and M7 in Figure 2. We initialize a local sizing constraint to limit the maximum sizing difference between two transistors to maintain a high DPAT of a power-balanced SABL gate design, and then size up the transistors. Through Monte Carlo simulations, we find that the optimized $\mu(LPAT)/\sigma(LPAT)$ is achieved when the leakage power of 00 pattern and 11 pattern are as similar as possible. As a result, the length of M3 and M6 should be sized up more than M4 and M7 within the global and local sizing constraints.

4.2 System-level Mitigation

At the system level, we aim to mitigate the degradation probability of MTD considering both dynamic power and leakage power. After applying the gate-level optimization, the degradation of MTD can actually become deteriorated. Since the system-level dynamic power has a linear dependence on the total circuit node capacitance, the nodes with sized-up transistors will consume extra amount of dynamic power and the MTD related the dynamic power could be affected. Therefore, we use a heuristic approach to resize some transistors connected to these nodes for a system-level DPAT optimization. If the resizing results still cannot meet the desired MTD, we have to reduce the local sizing constraint at the gate level. The entire design optimization procedure (shown in Figure 8) can take several iterations, and a simulation-based DPA is called to acquire the MTD as long as a new transistor sizing is done. For our DES component with less than thousands of gates, we perform four iterations to reduce the degradation probability of MTD to be 18%, which compensates for the non-optimized design by 40%. The design penalty of our optimization is 0.9% power consumption and 1.5% area overhead. The system performance remains almost the same. For a very large cryptosystem implementation, it is prohibitively expensive to perform too many optimization iterations. Thus, we recommend a divide-and-conquer strategy to optimize only one component of the entire cryptosystem at one time.

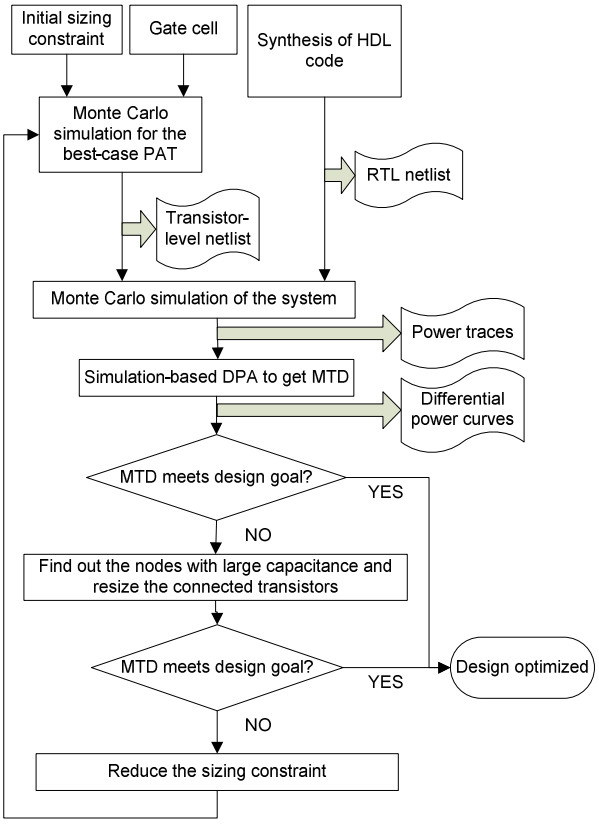


Figure 8. Design optimization procedure of mitigating process variation impacts on power-attack tolerance.

5. CONCLUSIONS

In this work, we analyze the process variation impacts on the power-attack tolerance (PAT) of embedded cryptosystems. We consider a power model with both data-dependent dynamic power and leakage power. Based on Monte Carlo simulations using 45nm CMOS device and process variation models, we demonstrate the negative impacts of process variations on PAT in both standard-cell CMOS and SABL implementation of a DES cryptographic component. Finally, we propose a transistor-level sizing method to mitigate the effect of process variations on PAT and the overall measurement to disclosure (MTD), with minimal design overhead.

Security evaluations of embedded cryptosystems are required at various abstraction levels, with different dominant factors to impact PAT and MTD. For example, the impact of device process variations on PAT could be much less than other impacts such as power supply variations and interconnect variations. However, device process variations are intrinsic parameters of all gates in a cryptosystem, and can be optimized on a netlist prior to the place-and-route design process. Our method for improving PAT can be extended to different design levels with appropriate power models and level-specific optimizations.

ACKNOWLEDGMENTS

We thank Sandip Kundu, Kevin Fu, the members of UMass Amherst VCSG group, and especially the anonymous reviewers for constructive suggestions. We also thank Yan Qi for editing the manuscript. This work was supported in part by the National Science Foundation under Grant CNS-0627529, Semiconductor Research Corporation Task 1595, and Intel Corporation.

6. REFERENCES

[1] P. Kocher, J. Jaffe, B. Jun, "Differential Power Analysis", CRYPTO, LNCS 1666, pp. 388-397, 1999.

[2] S. Mangard, "Hardware Countermeasures Against DPA – A Statistical Analysis of Their Effectiveness", CT-RSA, LNCS 2964, pp. 222-235, 2004.

[3] K. Tiri, I. Verbauwhede, "A digital design flow for secure integrated circuits," IEEE Transaction on CAD, vol. 25(7), pp. 1197-1208, 2006.

[4] K. Tiri, I. Verbauwhede, "Simulation models for side-channel information leaks," ACM/IEEE DAC, pp. 228-233, 2005.

[5] L. Lin, W. Burleson, "Leakage-based differential power analysis (LDPA) on sub-90nm CMOS cryptosystems", IEEE ISCAS, pp. 252-255, 2008.

[6] J. Tschanz, J.Kao, S. Narendra, R. Nair, D. Antoniadis, A. Chandrakasan, 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(11), pp. 1396-1402, 2002.

[7] K. Tiri, "Side-channel attack pitfalls," ACM/IEEE DAC, pp. 15-20, 2007.

[8] D. Agrawal, S. Baktir, D. Karakoyunlu, P. Rohatgi, B. Sunar, "Trojan detection using IC fingerprinting," IEEE Symposium on Security and Privacy, pp. 296-310, 2007.

[9] K. Tiri, I. Verbauwhede, "Securing Encryption Algorithms against DPA at the Logic Level: Next Generation Smart Card Technology," CHES, LNCS, vol. 2779, pp. 125-136, 2003.

[10] P. Yu, P. Schaumont, "Secure FPGA circuits using controlled placement and routing," ACM/IEEE CODES+ISSS, pp. 45-50, 2007.

[11] S. Mukhopadhyay, K. Roy, "Modeling and estimation of total leakage current in nano-scaled CMOS devices considering the effect of parameter variation," ACM/IEEE ISLPED, pp. 172-175, 2003.

[12] R. Rao, A. Srivastava, D. Blaauw, D. Sylvester, "Statistical estimation of leakage current considering inter- and intra-die process variation," ACM/IEEE ISLPED, pp. 84-89, 2003.

[13] International Technology Roadmap for Semiconductors, 2006, http://public.itrs.net.

[14] W. Zhao, Y. Cao, "New generation of predictive technology model for sub-45nm design exploration," IEEE ISQED, pp. 585-590, 2006.

[15] W. Mendenhall, and T. Sincich, "Statistics for engineering and the sciences," 5th edition, by Prentice Hall, 2007.

[16] S. Bhunia, S. Mukhopadhyay, K. Roy, "Process variations and process-tolerant design," IEEE VLSI Design, pp. 699-704, 2007.

[17] K. Takeuchi, T. Tatsumi, A. Furukawa, "Channel engineering for the reduction of random-dopant-placement-induced threshold voltage fluctuation," IEEE IEDM, pp.841-844, 1997.

[18] R. Brodersen, M. Horowitz, D. Markovic, B. Nikolic, V. Stojanovic, "Methods for true power minimization", ACM/IEEE ICCAD, pp. 35-42, 2002.

[19] P. Gupta, A. Kahng, P. Sharma, D. Sylvester, "Selective gate-level biasing for cost-effective runtime leakage control," ACM/IEEE DAC, pp. 327-330, 2004.