Power Balanced Circuits for Leakage-Power-Attacks Resilient Design

Basel Halak[1], Julian Murphy[2], and Alex Yakovlev[3]
[1]EEE Group, School of Electric and Electronic Engineering, Southampton University, UK (bh9@ecs.soton.ac.uk)
[2]Queens University, Belfast
[3]Schools of Electric and Electronic Engineering, Newcastle University, UK

Abstract **The continuous rise of static power consumption in modern CMOS technologies has led to the creation of a novel class of security attacks on cryptographic systems. The latter exploits the correlation between leakage current and the input patterns to infer the secret key; it is called leakage power analysis (LPA). The use power-balanced (m-of-n) logic is a promising solution that provides an answer to this problem, such circuits are designed to consume constant amount of power regardless of data being processed. This work evaluates the security of cryptographic circuits designed with this technology against the newly developed LPA. Two forms of LPA are investigated, one is based on differential power analysis (LDPA) and the other based on Hamming weight analysis (LHPA). Simulations performed at 90nm CMOS technology reveal that (m-of-n) circuits are totally resilient to LHPA and have a higher security level against LDPA than standard logic circuits.**

Index Terms**—Side Channel Attacks, Leakage Power, Security, Cryptography**

I. Introduction

The continuous scaling of semiconductor devices has led to a sharp increase in the leakage current, the latter is becoming the dominant contributor to total chip power consumption. At the 65nm technology node, the leakage power is in the order of half the chip power consumption and is planned to be an even greater fraction in successive semiconductor technologies [1]. This trend has led to the creation of a novel class of side channels attacks (SCA) on cryptographic circuits, which exploits the dependency of the leakage current of CMOS integrated circuits on their input patterns to deduce the secret key [2], they are called leakage power attacks (LPA). Several types of LPA's have recently been reported to be successful [3, 4]. The first is called leakage differential power analysis (LDPA); it is based on the correlation between a set of leakage power measurements and a selection function related to the key. A second form of LPA is based on the correlation between the Hamming weight of the inputs of crypto cores and their corresponding set of static power measurements [3]. The latter is going to be referred to in this paper as leakage-based Hamming weight power analysis (LHPA).

Both of types of LPA attacks have been successfully conducted on real circuit as outlined in [3, 4].

To the best of our knowledge, countermeasures for LPA attacks have not been proposed yet.

However, techniques to enhance the security of cryptographic systems against dynamic-power-based SCA have been extensively researched. Solutions applied at the architecture level aim to improve the security of cryptographic cores by concealing the correlation between their dynamic power traces and the secret key, this is done by noise insertion [5], random clock frequency [6], randomization of the instruction streams [7] or random insertion of dummy instructions into the execution sequence of the algorithm [8]. Yet over time, the attacks have evolved and become more and more effective, and SCA attacks were still feasible even in the presence of countermeasures [8, 9]. This has created a need for a new approach that tackles the root of the problem (i.e. information leakage) [10]. This led to emerging solutions at the circuit level which try not to create any side channel information instead of attempting to conceal or de-correlate the information leakage. One particular example of this trend is sense amplifier based logic SABL [11], the essence of this method is to create a cell library in which all gates always uses a fixed amount of power regardless of their input patterns. SABL completely controls the portion of the load capacitance that is due to the logic gate. The intrinsic capacitances at the differential in and output signals are symmetric and additionally it discharges and charges the sum of all the internal node capacitances. Although this approach was proved to be effective against SCA, it suffers from a major disadvantage, namely, the nonrecurring engineering costs of a custom designed cell library development. This makes the design process of SABL-based crypto systems lengthy and difficult, which increases its time-to market and may render this technique impractical.

This work proposes the use of power balanced (m-of-n) logic style to design cryptographic cores resilient to LPA. The design principles of power-balanced (m-of n) circuits was first outlined in [12]. The essence of this method is to make energy consumed per clock cycle independent of data being processed. This technique is compatible with the standard logic IC design flow [13, 14], which makes its integration in current industrial practice an easy task, hence the attractiveness of this method.

Previous work has shown that cryptographic cores designed using power-balanced (m-of-n) logic style demonstrate higher level of security against traditional SCA than those designed with standard CMOS logic [12, 14, 15].

This work will prove for the first time that the use (m-of-n) logic style to design cryptographic system can be an effective countermeasure against the newly developed leakage-based power analysis. To the best of our knowledge this is the first countermeasure to be proposed for such attacks.

This work only considers the (1-of-2) logic which is a sub-class of (m-of-n) circuits; however, the results can be extended to all other classes. All implementations used in this this paper of (1-of-2) logic are solely based on the design principles outlined in [12]. Our investigation reveals that high level of security for cryptographic systems against LPA can be achieved with this approach. This result is obtained through a thorough security validation at 90nm technology node by

means of circuit simulations and statistical analysis. The remainder of this paper is organized as follows. Section 2 reviews all known types of leakage power analysis and details the procedure of such attacks. Section 3 investigates the resilience of (1-of-2) cryptographic circuit against these forms of LPA attacks. Conclusions are drawn in section 4.

II. LEAKAGE POWER ATTACKS PRINCIPLES

1) Leakage Power Dependency on Input Patterns

The overall power dissipation of a CMOS cryptosystem is composed of a dynamic part associated with charging and discharging of capacitance; together with a static part associated with the leakage current. Leakage power is mainly dependent on physical parameters of transistor, temperature variations, and supply voltage fluctuations [16]; however, its dependence on input patterns becomes significant in sub-100nm technology [17]. This can be attributed to three major parts of CMOS leakage, namely:

1.1) Sub-Threshold Leakage

Sub-threshold leakage occurs when a transistor is in the off-state. The accurate description of sub-threshold leakage model can be found in the latest BSIM4 manual [18]:

$$I_{leak} = I_0 \left[1 - exp(\frac{V_{ds}}{v_t})\right] . exp\left[\frac{V_{gs} - V_{th} - V_{off}}{n v_t}\right] \quad (1)$$

$$I_0 = \mu \frac{W}{L} \sqrt{\frac{q\varepsilon_{si} NDEP}{2Q_S}} v_t^2 \quad (2)$$

In (1), v_t is the thermal voltage, V_{th} is the threshold voltage, V_{off} is the offset voltage, n is the sub-threshold swing parameter, and I_0 is the current given by (2). It can be observed that sub-threshold leakage has exponential dependence on bias voltage V_{ds} and input voltage V_{gs}.

1.2) Gate Leakage

Gate leakage is due to direct carrier tunneling between the gate of the transistor and the silicon under the gate oxide. It is composed of the tunneling current between gate and substrate (I_{gb}), the current between gate and channel (I_{gc}), and the current between gate and source/drain (I_{gs} and I_{gd}). As the thickness of gate oxide is scaled down below 3nm, gate leakage becomes significant, especially when a transistor is in the on-state with a large gate voltage.

1.3) Band-to-Band Tunneling (BTBT) Leakage

BTBT leakage is dependent on the bias voltage between the substrate and drain/source. It happens across the reverse biased p-n junction, so it is affected by the substrate doping profiles. BTBT leakage current is dependent on V_{db} and V_{sb}.

In figure 1, a four-terminal NMOS transistor with two input patterns. In each pattern, various sources of leakage are denoted. This illustrates the dependency of total leakage current on input patterns. The same phenomenon can also be observed in PMOS transistors. Such dependency is exploited to wage leakage-based power analysis on cryptographic systems [3, 4, 19]. The procedure of two forms of LPA will be explained in the next subsection.

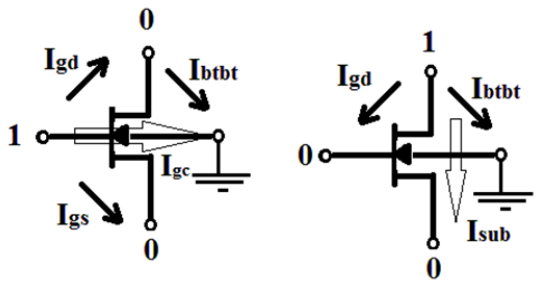

Figure 1.Leakage Dependency on Input Patterns of an NMOS

2) Leakage-Based Differential Power Attacks (LDPA)

The procedure of this attack is similar to that based on dynamic power measurements [4]. The leakage power profile has *m*k* data, denoted by $\{P_k^m\}$. The data are acquired by feeding *m* input plaintexts $\{I^1, I^2 \ldots I^m\}$ into the cryptosystem and measure their corresponding power dissipation. Within each input period, *k* power values are sampled including both leakage power and dynamic power. The main steps of this procedure are as follows:

Step1: Choose a selection function (*Y*) based on the structure of the circuit under attack. For example: for an S-box with 4 bit output {*S0*, *S1*, *S2*, *S3*}, the selection function can be associated with any of the 4 outputs. The value of a selection function is either 0 or 1.

Step2: Guess a secret key K, and compute the selection function (e.g. *Y =S2*(i)) for each input plaintext I^i .

Step3: Categorize the power profile $\{P_k^m\}$ to two groups based on the computing results of the selection function (*Y*). If Y=0, P_j^i are grouped to G0; others are grouped to G1.

Step4: For each guess of the key K, calculate Dj by (3), and the LDPA curve can be generated by k sample points D_1 to D_k.

$$D_J = \sum_{i \in G0} P_j^i - \sum_{i \in G1} P_j^i , j = 1,2,3 \ldots \ldots k \quad (3)$$

Step5: If K is the correctly guessed, a peak value will be displayed in the LDPA curve. Otherwise, the LDPA curve is compromised by uncorrelated power values.

Step6: Repeat the above steps for other guesses of the secret key until the correct key is found.

3) Leakage-Base Hamming-Weight Power Attack (LHPA)

LHPA exploits the correlation between the hamming weight of the input patterns of a circuits and its leakage power to reveal its secret key [2, 3, 19]. It is easier to wage than LDPA as it needs a fewer number of leakage power measurements. The procedure of this attack is as follows:

Step1: Choose an internal *m*–bit signal (*X*) that is physically generated within the cryptographic circuit under attack. In general, signal depends on both the input (*I*) and the secret key (*K*) of the cryptographic algorithm according to a well-defined function as follows.

$$X = f(I, K) \quad (4)$$

Where *f* is set by the cryptographic algorithm

Step2: Apply 2^m different input values I_i (with $i = 1, \ldots\ldots, 2^m$) and measure the corresponding leakage current $I_{leak,i}$ of the

cryptographic core at the point of time in which X is physically evaluated. In principle, this requires the knowledge of the clock period in which X is physically evaluated. The leakage measurements must be carefully performed. Indeed, once the input is applied to a logic gate, its leakage current is well known to have a transient variation and finally settles to the steady-state value after a period ranging from less than 1ns to a few tens of nanoseconds [20]. From these considerations, the clock period is generally comparable or greater than the period required to observe the steady-state leakage, particularly for nanometer technologies. As a consequence, usually, the adversary is not required to stop the clock to measure the leakage in the period in which is X physically evaluated. As a result of this step, an array $I_{leak,i}$ with size 2^m is obtained.

Step 3: the physical value of X is estimated for each possible input I_i according to (4). Since the generic input I_i is applied by the adversary, the only unknown variable in (4) is the secret key K ; hence, it must be guessed. For each possible guess of the secret key K_j (with $j=1\ldots\ldots$, 2^m), the resulting value of $X_{ij}= f(I_i, k_j)$ is found according to (4). As a result of this step, a two-dimensional array X_{ij} is found.

Step 4: the leakage current of the block generating X is estimated. This is done by calculating the Hamming weight $H(X)$, as in standard logic there is a linear relationship between the leakage current within the block generating X and its Hamming weight $H(X)$ [3]. The output of this step is a two-dimensional array with H_{ij} = $H(X_{ij})$ with ($i=1\ldots,2^m$) and ($j=1,\ldots\ldots\ldots,2^m$) and , which contains the Hamming weight of X for all applied inputs and key guesses.

Step 5: the measured leakage $I_{leak,i}$ and the estimated leakage H_{ij} are compared. For a given key guess K_j, the sequences sequences of inputs $I_{leak,i}$ and H_{ij} associated with the random (but known) sequence of inputs I_i (with $i= 1,\ldots\ldots\ldots,2^m$) can be thought of as random variables. When the key guess is correct (i.e. $K_j = K$), the estimated and measured leakages are maximally correlated. This means that the correct guess of K (i.e., the secret key) is that leading to the highest value of correlation coefficient ρ^j ($I_{leak,i,}$ H_{ij})among all possible guesses. Hence, the adversary must evaluate the correlation coefficients between the measured leakage and the Hamming weights for all key guesses, and identify the value of K that maximizes ρ and this will be the correct key. The output of this step is a 1-D array that contains correlation coefficient that corresponds to each key guess. In theory, the exact correlation coefficient cannot be evaluated since a finite number of samples of $I_{leak,i,}$ and H_{ij} are considered. Therefore in practical cases, Pearson' correlation coefficient is employed here, it can be calculated as follows.

$$r_j = \frac{\sum_{i=1}^{2^m} I_{leak,i}H_{ij} - 2^m \overline{I_{leak,i}}\,\overline{H_{ij}}}{(2^m-1)S_{I_{leak}}S_{H_{ij}}} \quad (5)$$

Where the sample means are calculated as follows:

$$\overline{I_{leak,i}} = \frac{1}{2^m}\sum_{i=1}^{2^m} I_{leak,i} \quad (6)$$

$$\overline{H_{i,j}} = \frac{1}{2^m}\sum_{i=1}^{2^m} H_{i,j} \quad (7)$$

And the sample standard deviations are calculated as follows.

$$S_{I_{leak,i}} = \sqrt{\frac{1}{2^m-1}\sum_{i=1}^{2^m}\left(I_{leak,i} - \overline{I_{leak,i}}\right)^2} \quad (8)$$

$$S_{H_{i,j}} = \sqrt{\frac{1}{2^m-1}\sum_{i=1}^{2^m}\left(H_{i,j} - \overline{H_{i,j}}\right)^2} \quad (9)$$

III. Secure Logic Design For Leakage Power Attacks

To establish how secure (1-of-2) logic is against LPA, we first need to find out how dependent the leakage power of the building blocks of this logic on their input patterns. Second, we need to evaluate the security of circuit designed using (1-of-2) logic against the newly developed LPA discussed in section 2.

1) Leakage power Dependence on Input Patterns

An experiment was constructed using cadence analogue simulator (*spectre*). Two-input standard logic gates (AND, OR and XOR) were simulated for all possible input patterns I = {00, 01, 10, 11} in 90nm technology. The corresponding leakage power was measured in each case. This experiment was repeated using the equivalent gates of (1-of-2)logic [12], in this case, the input patterns are I = {0101, 0110, 1001, 1010}

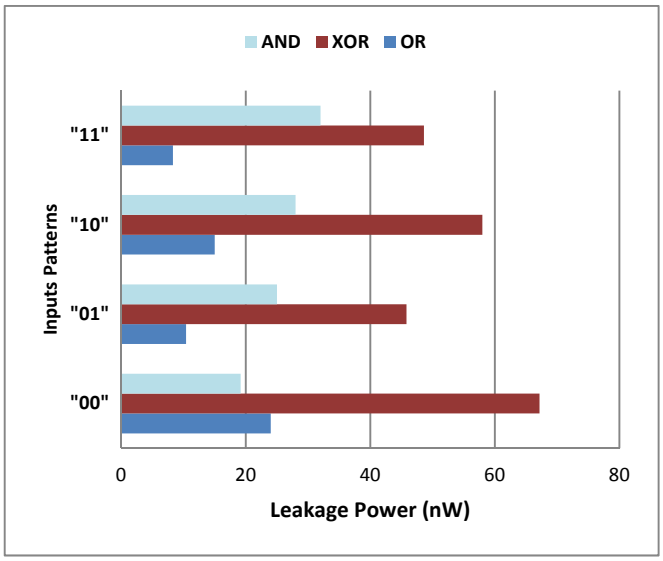

Figure 2: Standard Gates Leakage Power

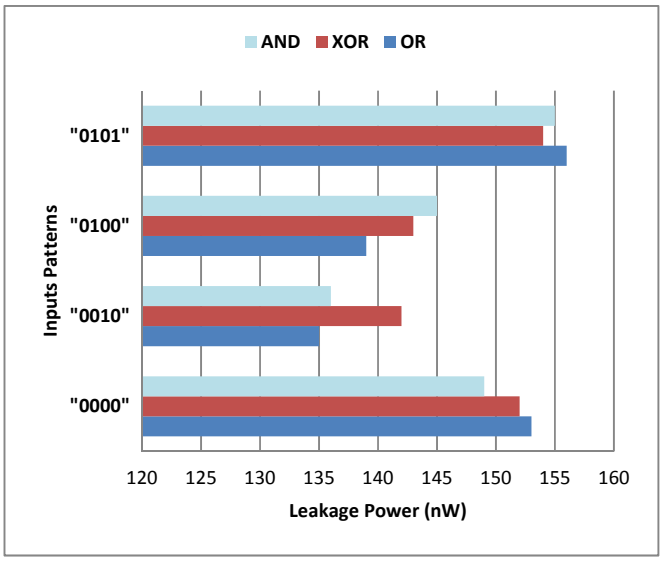

Figure 3: (1-of-2) Gates Leakage Power

The results depicted in figure 2 and 3 indicate that the leakage power of the basic gates of both standard and balanced logic styles is dependent on their input patterns. However, the dependency of (1-of-2) logic leakage on their inputs patterns is proved to be much weaker than that of standard logic leakage. To clearly demonstrate this we employ a statistical gauge called the coefficient of variation (*CV*). The latter is a normalized measure of dispersion of a probability distribution of the elements in a set *P* [21], It is calculated as the ratio between the standard deviation and the mean for the set of elements.

$$cv = \frac{\sigma}{\mu} \qquad (10)$$

In the case under consideration, the larger the coefficient of variation the larger the dependency of leakage power on input values. Figure 4 depicts the results for (AND, OR, XOR) gates designed using both logic styles. The results show that leakage power of (1-of-2) gates has less dependence on its input patterns than that of standard gates.

Another interesting aspect to investigate within this context is the temperature impact on the leakage-power–input-patterns dependency. Our Experiments showed that leakage power dependency on input patterns varies with temperature. However, this phenomenon is much less pronounced in (1-of-2) logic as shown in Figure 5 for AND gate.

These results suggest that circuits designed with (1-of-2) logic gates are more resilient to LPA than those designed with standard CMOS gates. This is especially true at high temperature where dependency of leakage on input patterns is much weaker in (1-of-2) circuits.

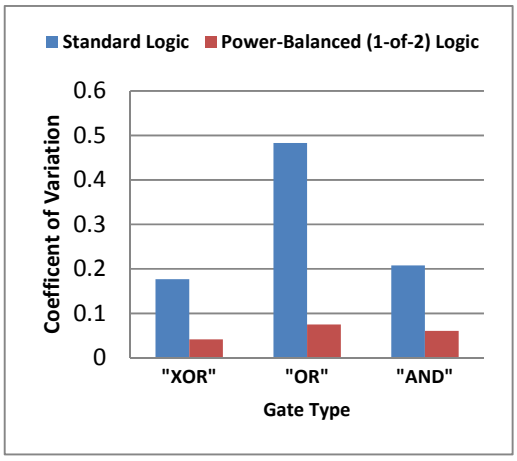

Figure 4: Coefficient of Variability Comparison

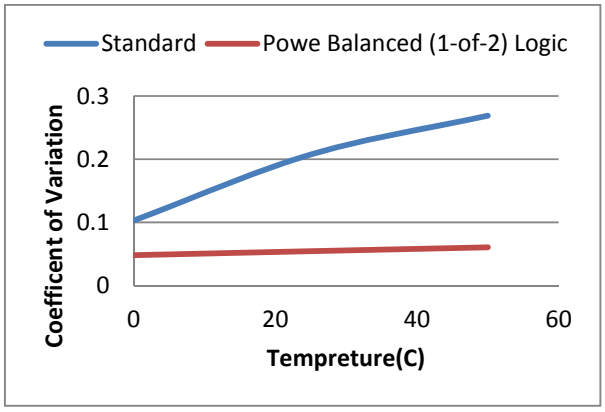

Figure 5: Impact of Temperature on Leakage Power Input Patterns Dependency

2) Security Evaluation of (1-of-2) Logic against LDPA

Two main methods can be employed for security evaluation. The first is to wage an attack and see whether or not the secret key can be guessed, in which case, the more secure the logic is, the more iterations the attack needs to be applied in order to find the correct key[15] ,this method is often unwieldy and time consuming . The second approach consists of employing a metric to measure how resilient the circuit is to LDPA, this is a more practical technique, and therefore it is going to be used in this work.

The basic component of symmetric key algorithms is the Substitution Box *(S-Box)*. In cryptographic cores, S-Boxes are typically used to obscure the relationship between the plaintext and the ciphertext. In general, an S-Box is a combinational mapping between an N bit input word and an M bit output word. For the purpose of this work a 4-inputs 4-outputs S-Box [22] was chosen for security validation.
The simple cryptographic core in figure 6 based on Serpent S-Box transformation was considered. Two versions of this cipher were constructed, one with standard CMOS gates (shown in figure 6) and a second version using (1-of-2) gates [12]. These circuits were implemented in Cadence environment using a 90nm CMOS process library.

The cryptographic core in figure 6 functions as follows: the plain word and the secret key are mixed in advance by XOR gates, and the result is ciphered by S-Box (a similar structure is observed in many other ciphers, such as DES and AES).
For the measurements of the standard logic-based cryptographic core, a 4-bit key is first XOR-d with a 4-bit plain words and the output is used directly as input to the standard S-Box. For the (1-of-2) implementation of the same core, the number of outputs and inputs is doubled as each single datum is represented with 2 bits. The design in both cases is fully synchronous.

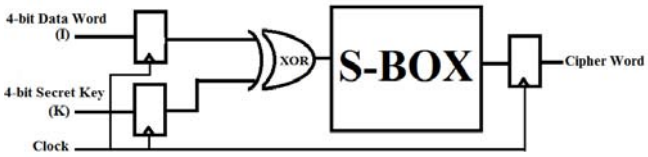

Figure 6: LPA Experimental Setup using Serpent-Based Cipher

To evaluate the security of the two version of this cipher, an evaluation metric based on the selection function is used. It is based on the point biserial correlation coefficient (PBC) [23]. This coefficient is a used when one variable is dichotomous, in this case 0 or 1. It ranges between the values 0 and 1.
To calculate PBC, assume that the dichotomous variable *Y* has the two values 0 and 1. If we divide the data into two groups, group 1 which received the value "1" on *Y* and group 2 which received the value "0" on *Y*, then the coefficient is calculated as follows:

$$PBC = \frac{M_0 - M_1}{S_n}\sqrt{\frac{n_0 n_1}{n^2}} \quad (11)$$

Here *Y* is the selection function, M_1 is the mean value on the continuous variable *X* (which is the leakage power in our case) for all data points in group 1(i.e. when the selection function value is 1), *and M0* is the mean value on the continuous variable *X* for all data points in group 2. Further, n_0 is the number of data points in group 1, n_1 is the number of data points in group 2 and *n* is the total sample size. S_n is the standard deviation used when you have data for every member of the population

$$S_n = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2} \quad (12)$$

The PBC equation is applied in this work to find a measure of security by using it to calculate the correlation for the different differential traces sampled.
In the case under consideration, the LDPA procedure explained in section 2 was performed on the circuit depicted in figure 6. The clock frequency was chosen to be 1 GHz, for each plain word, a 100 leakage power measurements were performed (k=100), the LDPA was repeated for 16 times (m=16). Differential power traces were obtained for all possible key guesses. The point biserial correlation coefficient was calculated for each trace. This experiment was repeated twice (for (1-of-2) and for standard logic implementation), the maximum value of PBC obtained in each case is used as a measure of comparison. The results are shown in Table I

Table 1: Point Biserial Correlation Coefficient Comparison

Standard Logic	0.59
1-of-2	0.299

The lower value of PBC for (1-of-2) logic implies a weaker correlation between leakage power and input patterns therefore implying a longer attack. Successful attacks above a PBC of 0.5 are very easy to conduct with a few hundred cipher texts. Successful attacks have been carried out at or just below a PBC of 0.3, however, It becomes more difficult to execute attacks significantly below this [24].

3) Security Evaluation of (1-of-2) Logic against LHPA

LHPA procedure described in section 2 was performed, the simulation setup was the same depicted in figure 6. Simulations in this case were carried out by exploring all possible combinations of plain words and keys and recording the leakage power. Pearson correlation coefficient was calculated for each key guess for the standard S-Box according to (5). A maximum value of 0.57 was obtained when the correct key is applied. It was not possible to calculate the Pearson correlation coefficient for the (1-of-2) S-Box, this is because its input patterns have always the same Hamming weight (H= 4). This trait makes it inherently resilient to LHPA. This is applicable to all circuit designed with (1-of-2) logic.

IV. Conclusions and Future Work

Leakage Power Analysis (LPA) poses serious threat to the information security of cryptographic systems in sub-100nm technologies. This is due to the rapid increase in leakage power at each new technology generation. This work has reviewed all recently developed forms of LPA. It has also proposed a Power balanced (1-of-2) logic as a promising countermeasure. This investigation showed that cryptographic circuit designed with this style is totally immune to LHPA and has a higher security level against LDPA than standard logic circuits. Our future will aim to experimentally wage LPA's on a fabricated (1-of-2) AES core to verify our results on silicon. We will also aim enhance the robustness of (1-of-2) circuit against LDPA by means of leakage power reduction and symmetric cell design.

References

1. International Technology Roadmap for Semiconductors (www.itrs.net).
2. Abdollahi, A., F. Fallah, and M. Pedram, Leakage current reduction in CMOS VLSI circuits by input vector control. IEEE Trans. Very Large Scale Integr. (VLSI) System, 2004. 12: p. 140-154.
3. Alioto, M., et al., Leakage Power Analysis Attacks: A Novel Class of Attacks to Nanometer Cryptographic Circuits. IEEE Transection on Circuit and Systems, 2010. 57(2): p. 355-367.

4. Lin, L. and W. Burleson, Leakage-based differential power analysis (LDPA) on sub-90 nm CMOS cryptosystems. n Proc. ISCAS, Seattle, WA, 2008. MAY: p. 252-255.
5. Paul Kocher, J.J., Benjamin Jun Diffrential power analysis. CHES 1999.
6. Akkar, M.L., Power analysis, what is now possible. ASIACRYPT, 2000.
7. Grabher, P., J. Großschädl, and D. Page. Non-deterministic processors: FPGA-based analysis of area, performance and security. in Proceedings of the 4th Workshop on Embedded Systems Security. 2009. Grenoble, France.
8. Clavier, C., J.S. Coron, and N. Dabbous, Differential power analysis in the presence of hardware countermeasures. Proceedings of the Second International Workshop on Cryptographic Hardware and Embedded Systems, 2000. 1965 LNCS: p. 252-263.
9. Kris Tiri, I.V., A VLSI Design Flow for Secure Side-Channel Attack Resistant ICs. Design, Automation and Test in Europe Conference and Exhibition., 2005. 3: p. 58-63.
10. Aigner, M., et al. Side channel analysis resistant design flow. in Circuits and Systems, 2006. ISCAS 2006. Proceedings. 2006 IEEE International Symposium on. 2006.
11. Tiri, K., M. Akmal, and I. Verbauwhede. A dynamic and differential CMOS logic with signal independent power consumption to withstand differential power analysis on smart cards. in Solid-State Circuits Conference, 2002. ESSCIRC 2002. Proceedings of the 28th European. 2002.
12. NewcastleUniversity, Cryptographic processing and processors. U.K. Patent Appl. No. 0719455.8, 2007.
13. Sokolov, D., et al., Design and analysis of dual-rail circuits for security applications. Computers, IEEE Transactions on, 2005. 54(4): p. 449-460.
14. Murphy, J. and A. Yakovlev. An Alternating Spacer AES Crypto-processor. in Solid-State Circuits Conference, 2006. ESSCIRC 2006. Proceedings of the 32nd European. 2006.
15. Burns, F., et al., Security Evaluation of Balanced 1-of-n Circuits. Very Large Scale Integration (VLSI) Systems, IEEE Transactions on, 2010. PP(99): p. 1-5.
16. Su, H., et al., Full chip leakage estimation considering power supply and temperature variations. ISLPED 2003: p. 78-83.
17. Rastogi, A., K. Ganeshpure, and S. Kundu, A study on impact of leakage current on dynamic power. ISCAS, 2007.
18. al., C.H.e., BSIM4.6.1 MOSFET Model. User's manual, 2007.
19. Giorgetti, J., et al., Analysis of data dependence of leakage current in CMOS cryptographic hardware. Proc. GSLVLSI, Stresa,, 2007: p. 78-83.
20. Narendra, S.G. and A. Chandrakasan, Leakage in Nanometer CMOS Technologies. Berlin, Germany: Springer-Verlag, 2006.
21. Hendricks, W.A. and K.W. Robey, The Sampling Distribution of the Coefficient of Variation. Annals of Mathematical Statistics, 1936. 7(3): p. 129-132.
22. R. Anderson, E.B., and L. Kundsen,, A proposal for the Advanced Encryption Standard, AES proposal http://www.cl.cam.ac.uk/ftp/users/rja14/serpent.pdf). 1998.
23. Olsson, U., F. Drasgow, and N.J. Dorans, The polyserial correlation coefficient. Psychometrika, 1982. 47(3): p. 337-347.
24. Aumônier, S., Generalized correlation power analysis. Ecrypt Workshop: Tools for Cryptanalysis, Kraków, Poland, 2007.