Dual-Rail Random Switching Logic: A Countermeasure to Reduce Side Channel Leakage[⋆]

Zhimin Chen and Yujie Zhou

Shanghai Jiao Tong University, China
chenzhimin@sjtu.edu.cn, zhou863@vip.sina.com.cn

Abstract. Recent research has shown that cryptographers with glitches are vulnerable in front of Side Channel Attacks (SCA). Since then, several methods, such as Wave Dynamic Differential Logic (WDDL) and Masked Dual-Rail Pre-charge Logic (MDPL), have been presented to make circuits clean. In this paper, we propose a more accurate power model based on logic gates' output transitions and divide it into pieces according to input signals' transformations. Based on our model, we demonstrate that 1-bit masked logic gates with asynchronous inputs always leak side-channel information from their output transitions. Therefore, even those gates designed without glitches are still susceptible to be attacked. To solve this problem, Dual-Rail Random Switching Logic (DRSL) is presented. By introducing a local pre-charge signal, DRSL gates have their inputs synchronized. Experimental results indicate that DRSL eliminates most of the leakage.

Keywords: Side Channel Attacks, DPA, Gate Level Masking, DRSL, Dual-Rail, Pre-charge.

1 Introduction

Until Paul Kocher *et al.* [1] proposed practical Side Channel Attacks (SCA) on chips, especially powerful Differential Power Analysis (DPA), people generally thought that cryptographic algorithms implemented in hardware chips were secure, therefore, they put more attention on security of protocols and mathematic algorithms. But since then, people began to pay more attention on implementations, and lots of countermeasures have been proposed in the last few years.

The earliest ways to act against DPA were called "Ad-hoc Approaches" [2], such as adding noises, randomizing execution sequence and so on. The drawback of this kind of countermeasures is that they do not prevent attacks completely: attacks can still be successful by taking more samples and signal processing.

For the purpose of preventing DPA completely, methods to protect cryptographers on the algorithm level were presented. Louis Goubin *et al.* [3] proposed

[⋆] This work has been supported by National Science Fund for Creative Research Groups (60521002) and Shanghai AM Fund (0425)

a way called duplication (or masking). Subsequently, masking method has been improved by many researchers [12, 13, 16, 17, 18].

On the other hand, more generic countermeasures are also under discussion. These countermeasures are on circuit level. We call them more generic in that they are not constrained to a certain cryptographic algorithm. Once a practical method is found, designers need not to care about the security of implementations for a specific algorithm. This makes possible the automatic design. These measures fall into two categories: complementary circuits and gate level mask circuits.

Kris Tiri and Ingrid Verbauwhede [7] proposed a complementary logic called "Sense Amplifier Based Logic" (SABL), in which "Dual-rail" and "Pre-charge" are employed. Considering SABL requires a new core cell library, "Simple Dynamic Differential Logic" (SDDL) and its refinement "Wave Dynamic Differential Logic" (WDDL) came into being afterward also under efforts of Kris Tiri [8]. Compared with SABL, WDDL only makes use of common cells.

Besides complementary circuits, masking on gate level is analyzed in [9], and implementation of masked gate circuits has been presented by Trichina and Korkishko in [10, 11].

Though the above methods, in both algorithm level and circuit level, aim at preventing DPA completely, they still leak side channel information. For masking methods, outputs' transitions of logic gates are dependent on the input signals when glitches exist [4]. What's more, in [5], Stefan Mangard *et al.* did a successful attack on masked AES hardware implementations with glitches. For complementary circuits, loading capacitance is hard to control for deep submicron process technologies where the transistor sizes and wiring widths continuously shrink [6].

To overcome the disadvantages of both masked and complementary circuits, Thomas Popp and Stefan Mangard in [6] bound masked and complementary circuits together and showed us "Masked Dual-Rail Pre-charge Logic" (MDPL). By absorbing "pre-charge protocol" and "Dual-Rail encoding", no glitches appears in MDPL circuits; by masking intermediate value with random bit, designers do not have to consider routing constrains.

However, we find that predictable energy dissipation still appears whenever inputs of a logic gate arrive at different moments, no matter glitches exist or not. This means that the previous methods are still susceptible to be attacked, including WDDL and MDPL. We did attack simulation with Hspice and the results demonstrate that our opinion is reasonable.

What should be mentioned is that Daisuke Suzuki *et al.* [15] also presented a kind of masked logic gate called "Random Switching Logic" (RSL). RSL belongs to Single-Rail circuits. All inputs to a RSL gate are synchronized by a pre-charge signal (called "enable signal" in [15]), but how to generate such a pre-charge signal was not mentioned yet. We think it is hard to generate such a pre-charge signal for each gate respectively in Single-Rail circuits.

In this article, we propose a power dissipation model according to a gate's output transitions, and divide it into pieces according to the input transitions. Based on our model, we demonstrate that 1-bit masked logic gates still leak side channel information. As an effective countermeasure, Dual-Rail Random Switching Logic (DRSL) is presented, in which inputs are synchronized for each gate respectively. Our experimental results show that DRSL reduces most of the side channel leakage. Therefore, DRSL is more robust than other logics.

This article is organized as follows. In Section 2, a mathematical model of power consumption and theoretical analysis of gate leakage are proposed. Our logic DRSL is presented in Section 3. Experimental results are given in Section 4.

2 Mathematical Models and Analysis

2.1 Gate Model

A logic gate in a cryptographer performs a Boolean algebra function. Factors that influence a gate's output values can be categorized into two groups: one is those determinable factors that can be decided by internal keys and outside input (or output) data; the other is the independent factors, such as the internal generated random numbers. For simplicity, we, here, only consider gates with only one output. What's more, for the practical consideration, each logic gate discussed in this article has only one independent factor. Then our model can be described in Equation 1.

$$q = f(a_0, a_1, \cdots, a_{n-1}, m) \tag{1}$$

where q is the output value; $a_0, a_1, \cdots, a_{n-1}$ are n factors related to key and outside data while m is the internal independent factor, f is the Boolean function that the gate performs. Hereafter, we also represent $a_0, a_1, \cdots, a_{n-1}$ as A for simplicity.

In a gate level masked circuit, 'm' is a mask signal, 'a_i' is the unmasked value of a masked input and 'q' is a masked output. A common digital circuit can be considered as a special subset of masked circuits, in which 'm' equals to a constant '0' or '1'.

2.2 Power Model

Power consumed by a CMOS gate is determined by many factors, such as output transition, load capacitance, self capacitance, clock frequency, supply voltage, and switch voltage [14]. In this article, we mainly focus on output transitions. We define the output transition as (q_{i-1}, q_i). Correspondingly, energy consumed can be defined as $E(q_{i-1}, q_i)$.

In a combinational circuit, input signals to a gate always arrive at different moments. The result following this is that outputs would probably switch several times during a clock cycle before they reach stable values. This is what we usually call "glitches". Suppose inputs arrive at k different moments, then power consumption can be represented as shown in Equation 2.

$$E = (E_0, E_1, \cdots, E_i, \cdots, E_{k-1}, E_k) \tag{2}$$

where E_i is the gate's power consumption during the input arriving intervals between moment i and moment *i+1*. When voltage of the output at moment i (v_i) and *i+1* (v_{i+1}) are both stable values (for example, 0v or 1.8v in 0.18μm technology), energy can be written as $E(0,0)$, $E(0,1)$, $E(1,0)$, or $E(1,1)$. Otherwise, if at least one of them is not stable, energy consumed can be represented as $tE(0,1)$ or $tE(1,0)$ by employing a coefficient 't' ($0 < t < 1$). Here, t is determined by v_i and v_{i+1}. From another point of view, t is mainly determined by the length of the interval, and is independent on the value of A.

2.3 Analysis

When attacking cryptographers using DPA, attackers aim to discover whether their key guesses are correct. Explaining this with our model, a correct key guess brings us a correct prediction of internal predictable factors, while incorrect key guesses lead to wrong predictions. If some statistical characteristic of the energy dissipated depends on the predictable factors, then attackers can make use of the power consumption as side-channel information to judge whether their key guesses are valid. Hence, secure cryptographers should have their power dissipation statistically independent on those predictable factors.

DPA can target on a circuit element (CE), which is a (group of) gate(s). Output values of a CE are statistically independent of others, so independence between the power consumption and the internal predictable factors lays on no correlation between E and A of a CE. What's more, we hold the opinion that independence between E and A at every time can be satisfied only if every element E_i of E is statistically independent on A, otherwise, the cryptographers would probably suffer from DPA.

In pre-charge circuits, at the beginning of evaluation phase, every signal has an initialized value: 0. (In some logics, signals are pre-charged to 1, but there is no essential difference.) As mentioned before, coefficient 't' is independent on A, hence, independence of E_i and A stands on independence between q_{i+1} and A ($q_i = 0$). This is the main topic of the following discussion.

Single-Rail Circuits In a Single-Rail circuit, each CE has only one output. The independence between q and A can be described in an equation as follows.

$$P(q = 0/A_i) = P(q = 0/A_j) \tag{3}$$

where P is the conditional probability, A_i and A_j are arbitrary sets of $(a_0, a_1, \cdots, a_{n-1})$. What's more, q must not be a constant and is related to every input.

Until now, the problem becomes to designing a logic gate that satisfies Equation 3 in all the k time intervals during a clock cycle. First, we consider the scenario that all inputs have arrived at this gate.

Lemma 1. *Let f be a logic gate's Boolean algebra function, q be its output and $a_0, a_1, \cdots, a_{n-1}$, and m be its n+1 independent variables: $q = f(a_0, a_1, \cdots, a_{n-1}, m)$. When q does not equal to constant 0 or 1, and is correlated to every input,* *then the necessary and sufficient condition for the statistical independence between q and $a_0, a_1, \cdots, a_{n-1}$ is*

$$q = f(a_0, a_1, \cdots, a_{n-1}, m) = g(a_0, a_1, \cdots, a_{n-1}) \oplus m \tag{4}$$

and

$$P(m = 0) = P(m = 1) = 1/2$$

where g is a Boolean algebra function; P is the probability. (Since lemmas in this article are easy to prove, we do not list their proof here.)

As we can see, to make circuits designed resistant to DPA, signals propagating inside should be masked as $a \oplus m$ or $\bar{a} \oplus m$.

When considering other cases, we take the kth interval as an example. In this interval, only one input has not arrived at the gate, which means either one of the masked signals ($a_i \oplus m$) or the masking signal (m) remains pre-charged.

If the last one is $a_i \oplus m$, we define the delayed signal as a_{im}. Since a_{im} is pre-charged to 0, we can assume that a_i equals to m in this interval. Then Equation 4 can be rewritten as follows.

$$q = f(a_0, a_1, \cdots, a_{n-1}, m) = g(a_0, a_1, \cdots, a_{i-1}, m, a_{i+1}, \cdots, a_{n-1}) \oplus m \tag{5}$$

Is q in this case still independent on the remaining predictable factors $(a_0, a_1, \cdots, a_{i-1}, a_{i+1}, \cdots, a_{n-1})$? According to Lemma 1, we should make sure whether there exists a Boolean algebra function h satisfying the following equation.

$$q = f(a_0, a_1, \cdots, a_{n-1}, m) = h(a_0, a_1, \cdots, a_{i-1}, a_{i+1}, \cdots, a_{n-1}) \oplus m \tag{6}$$

Lemma 2. *When a Boolean function f can be written as Equation 5, it cannot be rewritten into Equation 6.*

If the last one is signal m, we can represent output q with the same equation as before while replacing a_i with $a_i \oplus m$, and m with 0 (m is still pre-charged). So Equation 4 can be rewritten as follows.

$$q = g(a_0 \oplus m, a_1 \oplus m, \cdots, a_{n-1} \oplus m) \oplus 0 \tag{7}$$

Still, we should make sure whether there is a function h which satisfies Equation 8.

$$q = h(a_0, a_1, \cdots, a_{n-1}) \oplus m \tag{8}$$

Lemma 3. *when a gate's logic function can be described as Equation 7 and Equation 8, then n must be an odd number and*

$$h(a_0, a_1, \cdots, a_{n-1}) = f_a(a_0) \oplus a_1 \oplus \cdots \oplus a_{n-1} \tag{9}$$

According to Lemma 3, gates, such as masked AND and OR, do not satisfy Equations 7 and 8 simultaneously. Therefore, when m arrives last, output q is dependent on predictable factors A. Since AND and OR gates are the main components of cryptographers, so we can say that delay of the mask signal also has side channel leakage.

Based on Lemma 1 to Lemma 3, we can make a conclusion:

Conclusion 1. *In Single-Rail Circuits with all signals masked by the same random bit, when inputs arrive at logic gates at different moments, predictable factors dependent power dissipation appears no matter glitches occur or not. What's more, if inputs to a gate are pre-charged asynchronously, leakage would also occur.*

Dual-Rail Circuits As for the Dual-Rail Circuits, the independent circuit element is a pair of complementary signals. Therefore, Equation 4 should be rewritten as follows.

$$\begin{aligned}(Q_1, Q_0) &= q + \bar{q} = f(A, m) + \overline{f(A, m)} \\ &= g(a_0, a_1, \cdots, a_{n-1}) \oplus m + g(a_0, a_1, \cdots, a_{n-1}) \oplus \bar{m}\end{aligned} \tag{10}$$

where '+' represents common addition; q and $\bar{q}$ are a pair of complementary signals. $\bar{q}$ equals to the inversion of q in evaluation phase, while equals to q in pre-charge phase. Therefore,

$$Q_0 = q \oplus \bar{q}, Q_1 = q\bar{q}$$

For a Dual-Rail Circuit resistant to DPA, both Q_0 and Q_1 should be statistical independent on A.

Using the same proof methods employed in last section, we can demonstrate that when inputs to a gate arrive asynchronously, side-channel leakage occurs as well. Therefore, we can get Conclusion 2 as follows.

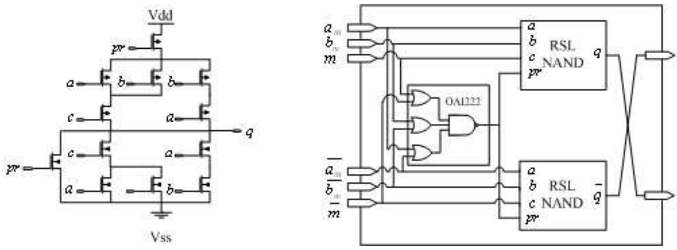
Conclusion 2. *In Dual-Rail Circuits with all signals masked by the same random bit, when inputs arrive at logic gates at different moments, predictable factors dependent power dissipation appears, no matter glitches occur or not. What's more, if inputs to a gate are pre-charged asynchronously, leakage would also occur.*

3 Dual-Rail Random Switching Logic

3.1 Basic Cells

Section 2 tells us that besides "free of glitches" and "no routing constrains", every internal gate in a DPA resistant cryptographer should have its inputs synchronized. DRSL is devised under such a guideline. To suppress glitches, "pre-charge" protocol is used; to remove routing constrains, random mask is introduced; to synchronize input signals, a local pre-charge signal is generated. The main idea of DRSL is derived from RSL and MDPL. But compared with MDPL, the advantage of DRSL is that it avoids side channel leakage caused by asynchronous inputs. As for RSL, DRSL makes use of Dual-Rail method to make practical the generation of the local pre-charge signal (called “enable” signal in RSL) for every gate.

The schematic of a two-input DRSL AND gate is shown in Fig. 1. Fig. 1(a) presents a single rail element; Fig. 1(b) describes a DRSL AND gate with a logic part (two Single-Rail elements) and a pre-charge generation circuit in it.



(a) (b)

Fig. 1. (a). RSL NAND schematic, (b). DRSL AND schematic

In DRSL circuits, there are two work phases alternating with each other: one is pre-charge phase, the other is evaluation phase. In the pre-charge phase, all signals, including mask signal m, are pre-charged to 0; while in the evaluation phase, pre-charge signal turns to be invalid after all inputs are evaluated values. Pre-charge of the whole circuit is done in a way of waveform: starting from registers, propagating through combinational logic gates and finally running back to registers. A global pre-charge signal is not suitable in that, between logic gates, their inputs arrive at different moments. This is similar to WDDL and MDPL, however, the difference is that each DRSL gate has its own pre-charge circuit. A DRSL gate is pre-charged at the time when one of the inputs turns to be pre-charged value, and enabled after all its inputs are evaluated values. Thus, DRSL gates do not suffer from asynchronous inputs.

In a Single-Rail circuit, pre-charged values and evaluated values can both be 0, so it is hard to judge when all inputs are evaluated values. On the other hand, pre-charged and evaluated values in Dual-Rail circuit do not have intersection: the former can only be (0, 0), and the latter belong to (1, 0) and (0, 1). This makes it possible to identify the time when all evaluated inputs have arrived. Based on the above consideration, Dual-Rail circuits are preferable in our logic. Once the pre-charge signal is generated, input signals are synchronized. This property of DRSL allows converting all kinds of logic gates to DRSL. For example, XOR, which is not a monotonic gate, is not used in MDPL and WDDL. But in DRSL, XOR is accepted. What’s more, since DRSL is Dual-Rail, an inverter can be implemented by just swapping its two complementary inputs. The same as mentioned in [15], odd-number-input XOR and XNOR function does not need a random signal input in DRSL.

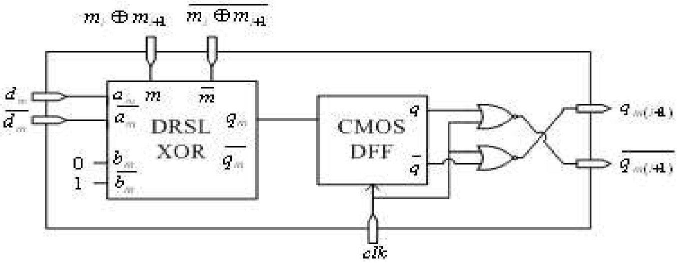


Fig. 2. DRSL D-flip-flop schematic

Since random mask changes every clock cycle, value stored in registers should be masked by the random signal for the following clock period. We incorporate the idea of MDPL D-flip-flop, in which a D-flip-flop consists of a RSL XOR gate, a common CMOS D-flip-flop and two CMOS NOR gates. Random signals for the XOR gate are $m_i \oplus m_{i+1}$ and $\overline{m_i \oplus m_{i+1}}$, where m_i is the random value for the current cycle and m_{i+1} is the one for the next. DRSL D-flip-flop schematic is presented in Fig. 2.

Table 1 compares DRSL cells in 0.18μm technology with the corresponding cells from TSMC 0.18μm standard cell library in area complexity.

Table 1. DRSL cells area complexity

DRSL Cell	Implementation	Area (gate equivalents)		Ratio
		DRSL	Standard	DRSL/std.
Inverter	Wire swapping	0	0.67	0
Buffer	2×Buffer	2.66	1.33	2
AND, OR(2-in)	2×RSL NAND, OAI	7.21	1.33	5.42
NAND, NOR(2-in)	2×RSL NAND, OAI	7.21	1	7.21
XOR, XNOR	2×RSL XOR, OAI	8.22	2.67	3.30
D-flip-flop	DRSL XOR, CMOS D-FF, 2×NOR	14.49	5.67	2.56

As can be seen from Table 1, DRSL AND, OR, NAND, and NOR gates cost much more area than standard gates. This is mainly caused by the local pre-charge circuit and the dual-rail circuit. However, as the gate becomes more complex, pre-charge circuit takes less proportion. Area ratio of DRSL XOR, XNOR, and D-flip-flop is smaller than DRSL AND and OR gates.

Compared with MDPL gates, DRSL AND (OR) gates cost more area than MDPL AND (OR) gates. But for XOR and DFF gates, DRSL costs less. Considering DRSL is compatible with MDPL, when designing DRSL circuits, a DRSL AND (OR) gate can be replaced by a MDPL AND (OR) gate if inputs to it are already synchronized.

3.2 Security Analysis

For every DRSL gate, outputs only change after all inputs arrive, energy elements before the last signal's arrival should be $2E(0,0)$, assume signals arrive at k different moments and the final output is q, then the last energy piece is $E(0,0)+E(0,q)$. Power consumption of a DRSL gate can be represented as follows.

$$E = (2E_0(0,0), 2E_1(0,0), \cdots, 2E_i(0,0), \cdots, 2E_{k-1}(0,0), E_k(0,0) + E_k(0,q))$$

Since output q is masked by a random signal, the above equation is not influenced by those predictable factors. So we can see the logic part of DRSL is free of leakage caused by asynchronous inputs.

Similarly, for the pre-charge circuit in DRSL, its power consumption can be described as follows.

$$E = (E_0(0,0), E_1(0,0), \cdots, E_i(0,0), \cdots, E_{k-1}(0,0), E_k(0,1))$$

Again, the equation is not related to those predictable factors, which means the pre-charge circuit is secure as well.

4 Experimental Results

We have performed DPA attacks simulation with Hspice on four 2-input AND gates implemented by common Single-Rail masked logic, WDDL, MDPL, and DRSL. All these gates are in 0.18μm technology. The layout parasitics have been neglected. Test circuits are illustrated in Fig. 3. In Fig 3(a), a_m arrives last; in Fig. 3(b), the random mask signal m arrives last.

For the Single-Rail masked AND gate, when a_m arrives later than b_m and m, then in the time interval, output q can be shown as follows.

$$q = ((a_m \oplus m)(b_m \oplus m)) \oplus m = ((0 \oplus m)(b_m \oplus m)) \oplus m = \bar{b}m$$

For WDDL and MDPL, we can also get the following results (m=0 for WDDL):

$$\bar{q} = ((\bar{a}_m \oplus \bar{m})(\bar{b}_m \oplus \bar{m})) \oplus \bar{m} = ((0 \oplus \bar{m})(\bar{b}_m \oplus \bar{m})) \oplus \bar{m} = \bar{b}\bar{m}$$

$$q_0 = q \oplus \bar{q} = \bar{b}, q_1 = q\bar{q} = 0$$

We simulate all the 8 possible combinations of input transitions on each of the AND gate. Current I(Vd)from circuits to power Vdd is the probed signal.

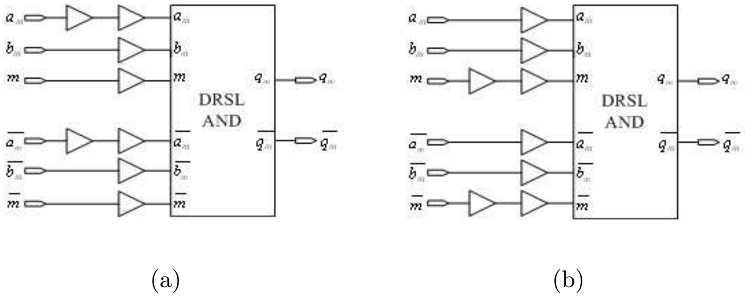


Fig. 3. (a). a_m arrives last, (b). m arrives last

Waveforms are divided into two groups, one with $b = b_m \oplus m = 1$, while the other with $b = 0$. Finally, we subtract the average of group 2 ($b = 0$) by the means of group 1 ($b = 1$) to get the difference. In the time interval when b_m and m have arrived and a_m is still pre-charged, only group 2 is possible to change output to be '1'; after a_m arrives, raise of output only occurs in group 1. So it is expected to get a figure with a valley followed by a peak in SRML, WDDL, and MDPL circuits. Results can be seen in Fig. 4.

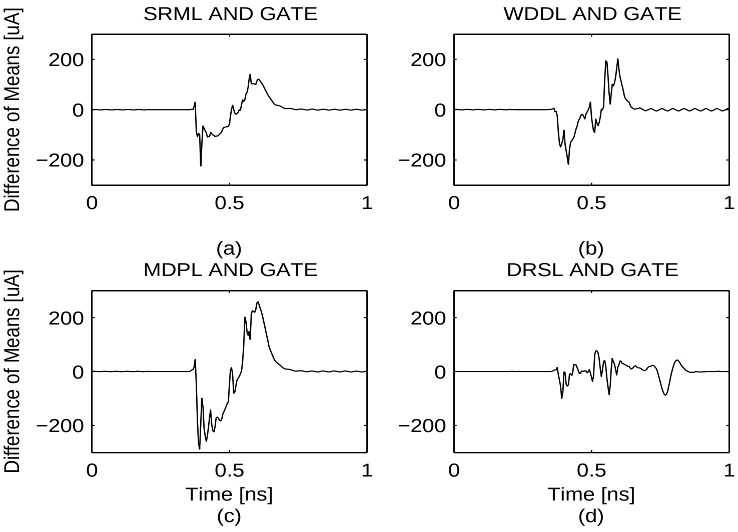


Fig. 4. Difference of means

When m arrives last, for Single-Rail masked AND gate:

$$q = ((a_m \oplus 0)(b_m \oplus 0)) \oplus 0 = (\bar{a}\bar{b}m) \vee (ab\bar{m})$$

For MDPL, we can also get the following results:

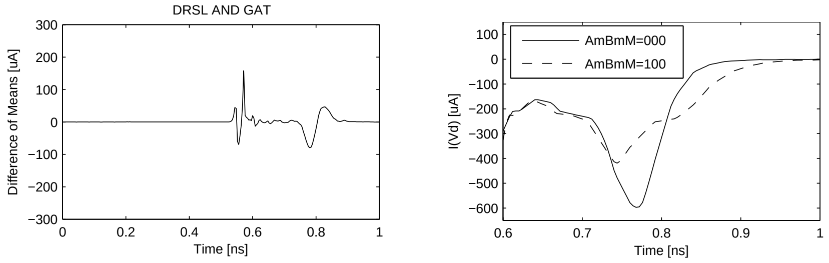
$$\bar{q} = ((\bar{a}_m \oplus 0)(\bar{b}_m \oplus 0)) \oplus 0 = (\bar{a}\bar{b}\bar{m}) \vee (abm)$$

$$q_0 = q \oplus \bar{q} = a \otimes b, q_1 = q\bar{q} = 0$$

In this case we divide waveforms of I(Vd) into two groups, one with $a = b$, while the other with $a\neq$b. Since this division happens to be the same as the former, their figures are similar (slight differences are caused by different self capacitance related to each input). We do not list the plots of this case here.

¿From Fig. 4 we can clearly notice the advantage of the DRSL AND Gate. The first three plots apparently have a valley followed by a peak, while the fluctuation of DRSL AND Gate is much smaller. Peak-to-peak values of each plot are approximately 418(SRML), 363(WDDL), 550(MDPL), and 117(DRSL) μA. Therefore, leakage of DRSL is reduced by at least 68%. When comparing the total power leakage, DRSL's performance is even better.

We also did an experiment in which every input reaches the gate at the same time. We divide the waveforms and get the difference of means in the same way as before. Result can be seen in Fig. 5(a). What's more, two immediate current I(Vd) plots ($a_m b_m m = 000$ and $a_m b_m m = 100$) are shown in Fig. 5(b).



(a)

(b)

Fig. 5. (a). Inputs synchronized, (b). Immediate Current

By comparing Fig. 4(d) and Fig. 5(a), we notice that the two plots are identical around 0.75ns, which means this part of leakage occurs even if inputs arrive at the same time. Accordingly, we divide the plot in Fig. 4(d) into two parts: the high-frequency fluctuation around 0.5ns and the comparatively low-frequency part near 0.75ns. We think the former be related to self capacitance. Leakage in this part is hard to identify. As for the latter, it is caused by different charging speeds. If $a_m = b_m = m$, all P transistors in the transiting RSL AND gate are open. This brings larger current and quicker change than other cases. In Fig. 5(b), charging current (-I(Vd)) belonging to $a_m b_m m = 000$ (real line) is larger than that of $a_m b_m m = 100$ (dotted line) at the beginning of transition. Since the stored charge is limited, the former also ends earlier than the latter. According to the above categorization, all traces belonging to $a_m = b_m = m$ were grouped into the second group ($b = 0$), so when subtracting the means of the two groups, a small valley followed by a peak appears. This kind of leakage is not considered in our model, as it does not come from the total power difference but the immediate power trace disagreement. Unfortunately, DRSL cannot avoid this kind of leakage. To minimize such kind of leakage is our job in the future.

5 Conclusion

We presented a power model where the power consumption of a logic gate depends on the value of the gate's output transition. Based on the model, we establish conditions for statistical independence between output transitions and the input values. Theoretical analysis shows that 1-bit masked gates with asynchronous inputs always leak side channel information. After that, we propose a kind of logic called Dual-Rail Switching Logic, which employs a local pre-charge circuit in each gate. Experimental results show that DRSL can eliminate most of the side channel leakage and therefore is more secure.

References

[1] Paul Kocher, Joshus Jaffe, and Benjamin Jun. *Differential Power Analysis.* In proceeding of Advances in Cryptology - CRYPTO '99, pp. 388-397, Springer, 1999.

[2] Suresh Chari, Charanjit S. Jutla, Josyula R. Rao, and Pankaj Rohatgi. *Towards Sound Approaches to Counteract Power-Analysis Attacks.* In proceeding of Advances in Cryptology - CRYPTO '99, pp. 398-412, Springer, 1999.

[3] Louis Goubin and Jacques Patarin. *DES and Differential Power Analysis - The "Duplication" Method.* In proceeding of Cryptographic Hardware and Embedded Systems - CHES '99, pp. 158-172, Springer, 1999.

[4] Stefan Mangard, Thomas Popp, and Berndt M. Gammel. *Side-Channel Leakage of Masked CMOS Gates.* In Topics in Cryptology - CT-RSA 2005, pp. 351-365, Springer, 2005.

[5] Stefan Mangard, Norbert Pramstaller, and Elisabeth Oswald. *Successfully Attacking Masked AES Hardware Implementations.* In proceeding of Cryptographic Hardware and Embedded Systems - CHES 2005, pp. 157-171, Springer, 2005.

[6] Thomas Popp and Stefan Mangard. *Masked Dual-Rail Pre-charge Logic: DPA-Resistance Without Routing Constraints.* In proceeding of Cryptographic Hardware and Embedded Systems - CHES 2005, pp. 172-186, Springer, 2005.

[7] Kris Tiri and Ingrid Verbauwhede. *Securing Encryption Algorithms against DPA at the Logic Level Next Generation Smart Card Technology.* In proceeding of Cryptographic Hardware and Embedded Systems - CHES 2003, pp. 137-151, Springer, 2003.

[8] Kris Tiri and Ingrid Verbauwhede. *A Logic Level Design Methodology for a Secure DPA Resistant ASIC or FPGA Implementation.* In Design, Automatin and Test in Europe Conference and Exposition (DATE 2004), IEEE Computer Society, pp. 246-251, 2004.

[9] Yuval Ishai, Amit Sahai, and David Wagner. *Private Circuits: Securing Hardware against Probing Attacks*. In proceeding of Advances in Cryptology - CRYPTO 2003, pp. 463-481, Springer, 2003.

[10] Elena Trichina. *Combinational Logic Design for AES SubByte Transformation on Masked Data*. Cryptology ePrint Archive (http://eprint.iacr.org/) , Report 2003/236, 2003.

[11] Elena Trichina and Tymur Korkishko. *Small Size, Low Power, Side Channel-Immune AES Comprocessor: Design and Synthesis Results*. In proceeding of the Fourth Conference on the Advanced Encryption Standard (AES), 2004.

[12] Elena Trichina and Tymur Korkishko. *Secure AES Hardware Module for Resource Constrained Devices*. In proceeding of Security in Ad-hoc and Sensor Networks: First European Workshop, ESAS 2004, pp. 215-229, Springer 2005.

[13] Elena Trichina and Lesya Korkishko. *Secure and Efficient AES Software Implementation for Smart Cards*. In proceeding of Information Security Applications: 5th International Workshop, WISA 2004, pp. 425-439, Springer 2004.

[14] A.P. Chandrakasan, S. Shen and R.W.Brodersen. *Low Power Digital CMOS Design*. In IEEE Journal of Solid State Circuits, Vol.27, N0.4. pp. 473-484, 1992.

[15] Daisuke Suzuki, Minoru Saeki, and Tetsuya Ichikawa. *Random Switching Logic: A Countermeasure against DPA based on Transition Probability*. Cryptology ePrint Archive (http://eprint.iacr.org/), Report 2004/346, 2004.

[16] Mehdi-Laurent Akkar and Christophe Giraud. *An Implementation of DES and AES, Secure against Some Attacks*. In proceeding of Cryptographic Hardware and Embedded Systems: CHES 2001, pp. 309-318, Springer 2001.

[17] Johannes Blomer, Jorge Guajardo, and Volker Krummel. *Provably Secure Masking of AES*. In proceeding of Selected Areas in Cryptography: 11th International Workshop, SAC 2004, pp. 69-83, Springer 2005.

[18] Elisabeth Oswald, Stefan Mangard, Norbert Pramstaller, and Vincent Rijmen. *A Side-Channel Analysis Resistant Description of the AES S-Box*. In proceeding of Fast Software Encryption: 12th International Workshop, FSE 2005, pp. 413-423, Springer 2005.