Overview of Dual Rail with Precharge Logic Styles to Thwart Implementation-Level Attacks on Hardware Cryptoprocessors

Jean-Luc Danger, Sylvain Guilley, Shivam Bhasin, Maxime Nassar, Laurent Sauvage

▶ **To cite this version:**

Jean-Luc Danger, Sylvain Guilley, Shivam Bhasin, Maxime Nassar, Laurent Sauvage. Overview of Dual Rail with Precharge Logic Styles to Thwart Implementation-Level Attacks on Hardware Cryptoprocessors. 2009. hal-00431261

HAL Id: hal-00431261
https://hal.science/hal-00431261

Preprint submitted on 11 Nov 2009

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Overview of Dual Rail with Precharge Logic Styles to Thwart Implementation-Level Attacks on Hardware Cryptoprocessors

— New Attacks and Improved Counter-Measures —

— Invited Paper at SCS'2009 —
Jean-Luc DANGER, Sylvain GUILLEY, Shivam BHASIN, Maxime NASSAR, Laurent SAUVAGE
Institut TELECOM, TELECOM ParisTech, CNRS LTCI (UMR 5141)
Département COMELEC, 46 rue Barrault, PARIS, FRANCE.
Email: < jean-luc.danger@telecom-paristech.fr >

***Abstract*—The security of cryptographic implementations relies not only on the algorithm quality but also on the countermeasures to thwart attacks aiming at disclosing the secrecy. These attacks can take advantage of the secret leakages appearing through the power consumption or the electromagnetic radiations also called "Side Channels". This is for instance the case of the Differential Power Analysis (DPA) or the Correlation Power Analysis (CPA). Fault injections is another threatening attack type targeting specific nets in a view to change their value. The major principle to fight the side-channel attack consists in making the power consumption constant. The Masking method allows the designer to get a power consumption which has a constant mean and a variance given by a random variable. Another manner is the Hiding method which consists in generating a constant power consumption by using a Dual-rail with Precharge phase Logic (DPL). This paper presents an overview of the various logic styles that have been promoted in the last six years, with an emphasis on their relative advantages and drawbacks.**

Keywords: Side Channel Attacks, Fault Attacks, Dual-rail with Precharge Logic (DPL), Differential Power Analysis.

I. Introduction

Many modern cryptographic algorithms are theoretically robust and immune from practical cryptanalysis in the "black box" model. However, some methods can be deployed to break the security by attacking the physical implementation of virtually any algorithm. These attacks can be mounted by merely observing or perturbing the targeted system. Observing the activity of the system and its correlation with potential guesses can yield sensible information. Such attacks are better known as Side Channel Attacks (SCAs). When a device is perturbed such that it yields a non-nominal output, this together with expected output can lead to the secret key. Such attacks are called Differential Fault Analyses (DFAs [1], [2]).

The advantages of SCAs are that the system is made to operate in its comfort zone. In such condition, it is difficult to detect that some devices may be observing the activity of the target. To defeat SCA efficiently, the countermeasures have at least to be submitted at the logic level. Dual-rail with Precharge Logic (DPL) is a class of countermeasures which aims at making the device activity constant and independent of data processed.

In this paper, we propose an overview of the main DPL styles with a focus on their vulnerabilities against both the Side-Channel and the Fault attacks.

The rest of the article is organized as follows. Section II presents the DPL countermeasure at logical level and the major vulnerabilities incurred by the backend. Then the Sec. III describes how the vulnerabilities are addressed by the already evaluated logic styles in either FPGA or ASIC. The Sec. IV explains how some optimizations and original solutions can be found using specific technologies. A synoptic comparison between the known logic styles is drawn in Sec. V. Finally, conclusions and perspectives are discussed in Section VI.

II. DPL Principle, Built-in DFA Resistance, and Latent Side-Channel Vulnerabilities

A. Information Hiding Rationale

The aim of dual-rail with precharge logic (DPL) is to hide the internal circuit's activity from a prospective attacker. If any sensitive variable update occurs with a constant activity, it is likely that all side-channels also are. Therefore, the measurable quantity from an attacker's point of view is independent from any secret value. The protocol of the DPL consists of two phases: precharge and evaluation. The precharge phase allows to start new computations from a known electrical state. It thus prevents unexpected transitions between two computation steps. The dual-rail signalization of the data is conveyed by two wires for each Boolean variable: $\texttt{NULL} = (0,0)$ or $(1,1)$ while in precharge and $\texttt{VALID} \in \{(0,1),(1,0)\}$ while in evaluation. Therefore, every evaluation consists in the transition of exactly one wire ($(0,0) \rightarrow (0,1)$ or $(0,0) \rightarrow (1,0)$). If the design is adequately balanced, which transition actually occurred is indiscernible by an attacker.

B. DPL Built-in DFA Resistance

Single bit faults are inefficient against DPL because they turn a VALID data into a NULL token, that propagates and leads to a non exploitable error since it hides the faulted value. This is the typical scenario described in the seminal paper [3], introducing the intrinsic immunity of DPL against some classes of DFA.

Highly multiple faults ($(1,0) \leftrightarrow (0,1)$) generate randomly a large quantity of NULL values along with some more unlikely but devastating bit-flips. However, as NULL values are systematically propagated, they proliferate very quickly after some combinatorial logic layers traversal. And as they have the nice property to contaminate VALID values, the risky coherent bit-flips (simultaneous $0 \xrightarrow{*} 1$ and $1 \xrightarrow{*} 0$ in one dual-rail couple) has a great chance to be jammed through the propagation towards the algorithm output. This absorption property is all the more efficient as the number of NULL generated by the multiple faults is high. Therefore, the only way to inject a poisonous fault is to stress the circuit sufficiently enough to have multiple faults, without nonetheless creating too many faults so as to leave a chance for them not to be absorbed during their percolation towards the outputs.

C. Vulnerabilities w.r.t. Side-Channel Attacks

Although perfectly sound at logical level, DPL ends up to be concretely implemented in physical devices. Now, the logical description of DPL ignores any timing and capacitance's notions.

Regarding the timing, three unbalanced behaviors can occur. On the way from the precharge to the evaluation, and vice-versa, there can exist:

1) spurious transitions, referred to as glitches, that negate the hypothesis of activity constantness, and
2) early evaluation effects. It takes place if the gate switching depends on the difference between the arrival time of the inputs.
3) technological bias. This flaw comes from the imbalance between the dual signals. It can be caused by the manufacturing dispersion, the place-and-route stage or merely the types og gate driving the true and false networks. This biais could be exploited by an attacker who measures the signal emanating from one wire of a pair.

The cross-coupling is another issue, still unexploited but probably also endangering the security of DPL designs.

The next section III studies how these latent flaws have been addressed by some existing DPL logics, whereas the section IV illustrates technology-dependent optimizations or innovative solutions.

III. DPL Families Based on Standard Cells

A. WDDL

Wave Dynamic Differential Logic (WDDL [4]) meets all the logical constraints of a DPL. The initial state is propagated by a wave of $(0,0)$ couples through the netlist thanks to the use solely of positive gates. The fact that exactly one half of the gates evaluate results from the duality between the true and false networks. In addition, the positivity of WDDL ensures the absence of glitches in the complete netlist. Notice that WDDL with gates propagating the NULL spacer but without being positive is easily broken in practice, as explained in [5]. However, as shown in [6], [7], WDDL is prone to early evaluation and early precharge. The Early Evaluation (EE) effect comes from the difference of delay between two variables of a same gate. Fig. 1 illustrates the EE flaw when variable a is in advance to variable b. In this case the output does not switch at the same time.

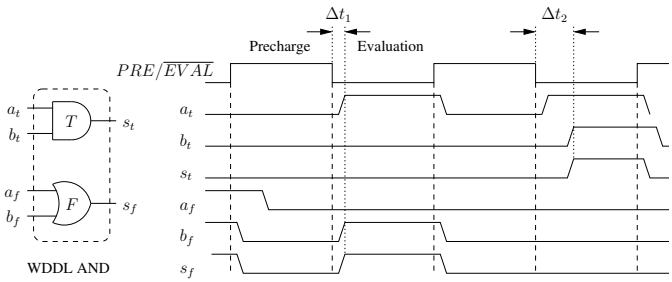

Figure 1. WDDL AND gate with the Early Evaluation flaw.

Moreover, the dual networks are not necessarily balanced, since the transistor structure of $x \mapsto f(x)$ and of $x \mapsto \overline{f(\overline{x})}$ differ. Those two issues have made possible some attacks on WDDL circuits, as described for instance by the authors of WDDL themselves in an ASIC [8] or independently in an FPGA [9]. Therefore, either incremental improvements or radically novel strategies have shown up.

B. MDPL

Masked Dual-rail with Precharge Logic (MDPL [10]) is an attempt to fix the otherwise imbalance of WDDL. The assumption is that, in some conditions, it can be difficult to constrain a router to balance the differential interconnect. Indeed, the two solutions available in the literature, namely the fat wire [11] and the backend duplication [12] methods, apply primarily to ASICs. The transposition to FPGA is possible, albeit with less fine-grain control over the result [13]. For this reason, MDPL proposes to swap the true and the false routes randomly, so as to emancipate from the fatal routing unbalance. By the same token, it makes up for the structural unbalance of the dual pair of gates. The only gates involved in the logic are majority functions, both for the true and the false networks. Nonetheless, MDPL fails to provide a solution to the early evaluation and precharge of WDDL.

C. DRSL

The primary focus of Dual-rail Random Switching Logic (DRSL [14]) is to make the evaluation and the precharge gates data-independent. For this reason, one pairwise unanimity gate[1] computes the validity of all inputs prior to allowing the

[1]The pairwise unanimity Boolean gate performs the following computation: $(x_\mathrm{T}, x_\mathrm{F}, y_\mathrm{T}, y_\mathrm{F}, \ldots, z_\mathrm{T}, z_\mathrm{F}) \mapsto (x_\mathrm{T} + x_\mathrm{F}) \cdot (y_\mathrm{T} + y_\mathrm{F}) \cdot \ldots \cdot (z_\mathrm{T} + z_\mathrm{F})$.

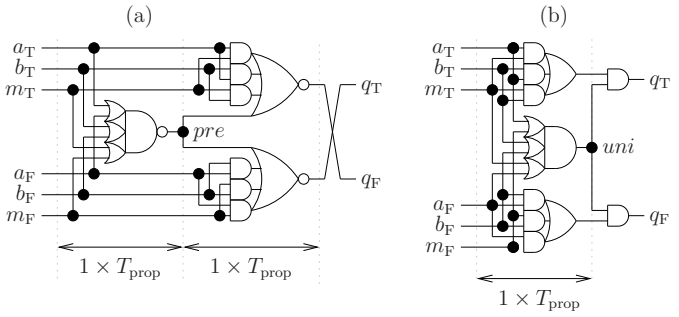

Figure 2. (a) Genuine DRSL AND gate and (b) Glitch-free variant.

gate from delivering any result, thus avoiding the EE flaw. On the contrary, the unanimity makes it possible for the overall DRSL logic to always anticipate the precharge. However, in the original design of DRSL, the functions are not required to be positive. The example of the AND function is sketched in Fig. 2(a). Hence the presence of data-dependent glitches in the return to precharge phase.

We carried out an extensive simulation of the DRSL AND gate when it returns to precharge. It happens that the DRSL AND gate glitches iff $a \oplus b = 1$, irrespective of the mask value.

Two solutions can be imagined to patch the glitching problem of DRSL. The first one consists in adding buffers to delay the signals so as to balance the paths within the DRSL gate. Another option consists in implementing DRSL in positive logic, as shown in Fig. 2(b). This solution has a cost in CMOS logic, because inverting gates are smaller than non-inverting ones (actually realized in practice by the composition of an inverting gate with an inverter [15]). However, this is not constraining in FPGA. A loss in area is nonetheless expected, as the functionality can only consist in positive gates, thereby limiting the degree of freedom of the the logic synthesizers. In this case, the new logic, that we name DRSL+, consists in MDPL augmented with a synchronization by an unanimity cell.

Another attack against DRSL is presented in [16]. Actually, this attack puts forward a vulnerability that is common to all masked DPL styles. The idea is that the masking of the gates allows to make up for the routing unbalance. However, the mask signal is itself differential and therefore unbalanced. As it is not balanced (since this is the hypothesis when resorting to masked DPL), it paradoxically opens the door to an attack on itself.

D. STTL

Secure Triple Track Logic (STTL [17]) eludes any glitching risk by waiting to evaluate and to precharge until all the inputs are either valid or NULL. This incurs useless delays in the return to precharge phase, which is however only detrimental to performance, not to security. The main drawback of STTL is the requirement to route one synchronization signal slower than the dual-rail, while granting a balanced routing within the dual-rail pair. However, the known methods to balance signals (fat wire and backend duplication) operate on a full netlist, and are therefore difficult to adapt on heterogeneous netlists, in which single-ended and dual-rail signals are mixed up.

E. BCDL

Balanced Cell-based Differential Logic (BCDL) [18] improves on STTL by accelerating the precharge phase, thanks to a global signal. As BCDL design allows to squeeze the precharge step, BCDL can compute about 80% faster than DRSL because the precharge is global.

Furthermore, as the global signal is, by design, faster than data signals, BCDL is free from the flaw identified in DRSL. Additionally, BDCL is a truly differential logic. BCDL and STTL can be seen as equivalent at the netlist level: input synchronization logic (either C-elements or unanimity gates) can be factored in STTL to the detriment of the systematic addition of a third routing resource.

F. WDDL Variants

Some variants of WDDL have also been devised to ease the balance of the WDDL networks. However, as already explained in the subsection devoted to MDPL, it is known that balancing the WDDL interconnect does not solve the early evaluation (EE) inherent to this logic. Nevertheless, we introduce them here because some of these logics have unexpected positive side-effects on their security w.r.t the EE.

1) DWDDL: Double WDDL (DWDDL) is introduced in [19] to counterbalance one unbalanced network with a dummy dual one. Although this solution is sound in theory, other efforts have been deployed to reduce the overhead associated with the further duplication of hardware in DWDDL. In [13], the design of a substitution box (sbox) in WDDL, similar to the WDDL in BDD-style presented in [20], allows for a separation between the true and false halves, thus allowing for a copy-and-paste of the two halves, that are thus guaranteed to have the same backend.

2) WDDL with Divided backend duplication: Divided backend duplication [21] attempts to go one step further by being applicable to any kind of logic (not only the sboxes), has roughly speaking the same overhead as WDDL, while being completely separable in the meantime. Basically, the true/false separation is achieved by preventing the inversions to be replaced by dual wires crossing. However, in a view to keep the precharge propagation to the NULL state, the inverters can be inhibited when in precharge: they are implemented as XNORs with the precharge signal. However, this alteration comes at the cost of two vulnerabilities insertion. If the precharge is concomitant to the clock, then glitches are going to occur due to races between the signals in a non-positive logic. If the precharge is asserted in an individual clock period, then the precharge state does not guarantee anymore a constant number of toggles at evaluation stage.

3) IWWDL: Eventually, Isolated WDDL (IWWDL) [22] is a different strategy to separate a WDDL netlist. Here, inverters are kept but potential glitches are stopped by systematically inserting one register after it. This strategy is expensive

in terms of area and requires a redesign of the controller. Additionally, the design becomes much more pipelined, which requires much higher clock frequencies to maintain an acceptable throughput. However, the benefit of this approach is to stop also the propagation of the EE wave. Apart from the very poor performance of IWDDL, this method is however very strong from a pure security standpoint. Only one point is questionable: isn't the complete separation of the netlist opening the door to well located EMA attacks, that can record selectively the activity from only one half of the netlist, thus defeating the activity constantness property. This issue is all the more stringent as the netlist is much larger in IWDDL than in WDDL, because of the large quantity of registers added for the pipeline.

4) WDDL w/o EE: WDDL w/o EE is a logic style dedicated to FPGA that removes the EE without computing a rendezvous. Instead, each functional half gate receives the true and false inputs, and decides to output the VALID value only when all the inputs are VALID. This behavior can be achieved by a purely combinatorial gate [23]. The detailed rationale behind the "WDDL w/o EE" style is the following:

- The gate outputs NULL$\{0,1\}$ when the inputs are NULL$\{0,1\}$ or transitional from this value.
- The gate outputs VALID only when all the inputs are VALID.
- In case of inconsistent values w.r.t. the DPL convention, the gate outputs an arbitrary NULL value.

This logic does not evaluate early by design, and propagates errors: if any input is stuck to NULL or if the input is out of specifications, then the output always remains to NULL too. In addition, this logic does not generate glitches even if the functionality is not positive, and can be inverting. Therefore, the synthesis is more optimized than for plain WDDL.

IV. Technological Specific DPL Styles

A. Full Custom Optimizations

In 2002, Kris Tiri introduces the "Sense Amplifier Based Logic" (SABL) logic style [24], [25], which aim is to make power consumption independent of both the logic values and the sequence of the data. It is therefore the first DPL proposal. Its principle consists in combining Differential and Dynamic Logic (DDL) like in the "Dynamic Cascode Voltage Switch Logic" (DCVSL) style, while fixing second order asymmetry in the gate (especially for complex logic functions), due to parasitic capacitance [26]. This allows to decorrelate the power consumption from the inputs. In 2006, Marco Bucci *et al.* [27] show that the balance of DPL gates can be improved by adding a systematic discharge after the evaluation. The resulting computations are thus based on a ternary pace: (1) pre-charge, (2) evaluation and (3) post-discharge. When applied to SABL, simulations reveal that a gain of two-order of magnitude is obtained in terms of balance.

SecLib is a full-custom logic style depicted in Fig. 3 introduced in 2004 by Sylvain Guilley *et al.* [28]. This logic is based on an quasi-delay insensitive asynchronous primitives,

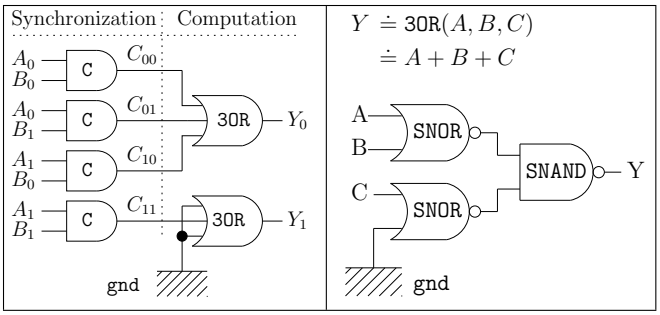

Figure 3. Schematic of the QDI secured (*aka* SecLib) AND gate (*left*) and its internal 3OR architecture (*right*).

that are balanced to provide constant evaluation and precharge time and dissipation. Specially crafted transistor-level symmetry grants SecLib a higher resistance level to attacks than WDDL, albeit at a high cost in terms of silicon area [29], [30], [31].

In [32], Loïc Duflot *et al.* describe an optimization for SecLib. The core idea, detailed by Fabien Germain's PhD [33], is to balance the computation thanks to conflict logic after an input configuration decoding stage.

In 2005, SABL and "Dynamic Current Mode Logic" (DyCML) [34] are compared by François Macé *et al.* [35]. In DyCML, only one of the output nodes is discharged during the precharge phase. This leads to better performances, such as a reduction by 80 % of the power delay product and by 50 % of the power consumption. In addition, DyCML is assessed to be more resistant to DPA than SABL.

Recently, Francesco Regazzoni *et al.* explore the resistance of "MOS Current Mode Logic" (MCML) against DPA [36], [37], [38] up to simulated attacks. Preliminary results show that MCML has a strong potential for protecting circuits.

B. Asynchronous Logic

Some asynchronous logic styles operate in a DPL mode. If the netlist and their layout is additionally balanced, asynchronous styles can be a candidate for secure computing [39], [40], [41]. In addition to the protection against side-channel attacks, asynchronous logics are also more tolerant to the environmental variations, which makes them inherently more difficult to attack with faults injections.

C. Reversible Differential Logic

Reversible logic is a means to compute without loosing energy at any step. This implies that any moment of the computation, the operations may be reversed. Two precursors in this field of research were Tommaso Toffoli [42] and Edward Fredkin [43].

They proved that the concept of reversible computing was indeed realizable physically, provided that the function to implement is logically reversible. Basically, they demonstrated that any bijection can be mapped onto a reversible physical system. However, two difficult issues were left uncovered by their works:

1) a generic synthesis method for arbitrary bijections, and also an algorithm to provide the most compact netlist, is still be found, and
2) an integrable electronic system suitable for the implementation of reversible logic is laking. Indeed, the only concrete example illustrating Toffoli & Fredkin's work was the famous albeit unpractical "Billiard Ball" model, that cannot extend to thousands of interactions, as required by our modern computational needs.

The first question has received some answers [44], [45]. Regarding the second point, it has been covered by some researchers, for instance in this article [46]. In this paper, the authors describe some implementations in CMOS for representative reversible logic gates.

V. DPL styles Comparison

Table I draws up a comparison of the main DPL styles, in terms of principle, design constraints and performance, highlighting most of the known advantages (masking, synchronization) and drawbacks (primitives and back-end constraints, and technological bias) of such countermeasures.

Masking allows to greatly reduce the technological bias, but also results in a significant increase of area. As a matter of fact, it requires at least a transformation of 2-input operations into 3-input majority function (MDPL) or into a 4-input RSL gate (DRSL).

Synchronization on both precharge and evaluation is mandatory to avoid glitches and early propagation effects.

Primitive constraints induce a higher complexity, by reducing the panel of usable functions (like in WDDL where only positive functions are allowed), or by binding the designer to use specific functions that can be more area-consuming or slower than basic ones (Seclib, MDPL, DRSL).

Back-end constraints generate extra design work as the P/R stage has to meet specific requirements to achieve a good balance between the T and F networks. It can also cause a loss of performance, like in STTL where the synchronisation signal must be manually made slower than the others, by adding delay elements between each gates, in order to ensure that it always switches last.

Technological bias corresponds to the imbalance between the True and False networks. It encompasses the load, interconnect and CMOS structure differences. This is a significant source of information leakage, and must therefore be as low as possible to ensure a perfectly secure countermeasure.

VI. Conclusions

In this article, we presented the different DPL logic styles aiming at Hiding the cryptoprocessors activity to thwart the side-channel attacks. Although the DPL logic is based on an elegant manner to obtain secure implementations, flaws exist at logical and physical level. The different logic styles are more or less able to counteract these negative effects but often with an higher complexity or back-end design. This paper permits the understanding of the main DPL style and draws a comparison between them in order to help the pros and the cons analysis. Research on new DPL styles is still active to improve the robustness and keep a good compromise with complexity and performances requirements.

Table I
DPL performance and security features overview.

Logic	Mask	Synchro		Constraints		Tech Bias	Speed
		Pre	Eval	Primitives	Back-end		
WDDL	no	✗	✗	positive funct only	balanced place&route	high	$< 1/2$
MDPL	yes	✗	✗	MAJ †	no	no	$< 1/2$
STTL	no	✓	✓	no	delay on sync signal	very low	$< 1/4$
DRSL	yes	✓	✗	no	no	no	$< 1/2$
Seclib	no	✓	✓	specific lib	back-end duplication	very low	$< 1/2$
IWDDL	no	✓	✓	no	netlist post-processing	low	$< \frac{1}{2 \cdot n_i}$ ‡
BCDL	no	✓	✓	no	balanced place&route	low	$> 1/2$

† MAJ stands for the majority gate: $MAJ(a,b,c) \doteq a \cdot b + b \cdot c + c \cdot a$.
‡ n_i is the maximum number of inverters amongst all combinatorial paths.

References

[1] E. Biham and A. Shamir, "Differential Fault Analysis of Secret Key Cryptosystems," in *CRYPTO*, ser. LNCS, vol. 1294. Springer, August 1997, pp. 513–525, Santa Barbara, California, USA.

[2] G. Piret and J.-J. Quisquater, "A Differential Fault Attack Technique against SPN Structures, with Application to the AES and Khazad," in *CHES*, ser. LNCS, vol. 2779. Springer, September 2003, pp. 77–88, Cologne, Germany.

[3] N. Selmane, S. Bhasin, S. Guilley, T. Graba, and J.-L. Danger, "WDDL is Protected Against Setup Time Violation Attacks," in *FDTC*. IEEE Computer Society, September 6th 2009, pp. 73–83, In conjunction with CHES'09, Lausanne, Switzerland. DOI: 10.1109/FDTC.2009.40; Online version: http://hal.archives-ouvertes.fr/hal-00410135/en/.

[4] K. Tiri and I. Verbauwhede, "A Logic Level Design Methodology for a Secure DPA Resistant ASIC or FPGA Implementation," in *DATE'04*. IEEE Computer Society, February 2004, pp. 246–251, Paris, France.

[5] S. Guilley, L. Sauvage, J.-L. Danger, T. Graba, and Y. Mathieu, "Evaluation of Power-Constant Dual-Rail Logic as a Protection of Cryptographic Applications in FPGAs," in *SSIRI*. Yokohama, Japan: IEEE Computer Society, jul 2008, pp. 16–23, DOI: 10.1109/SSIRI.2008.31, http://hal.archives-ouvertes.fr/hal-00259153/en/.

[6] D. Suzuki and M. Saeki, "Security Evaluation of DPA Countermeasures Using Dual-Rail Pre-charge Logic Style," in *CHES*, ser. LNCS, vol. 4249. Springer, 2006, pp. 255–269, Yokohama, Japan. http://dx.doi.org/10.1007/11894063_21.

[7] ——, "An Analysis of Leakage Factors for Dual-Rail Pre-Charge Logic Style," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E91-A, no. 1, pp. 184–192, 2008, DOI: 10.1093/ietfec/e91-a.1.184.

[8] K. Tiri, D. Hwang, A. Hodjat, B.-C. Lai, S. Yang, P. Schaumont, and I. Verbauwhede, "Prototype IC with WDDL and Differential Routing – DPA Resistance Assessment," in *Proceedings of CHES'05*, ser. LNCS, LNCS, Ed., vol. 3659. Springer, September 2005, pp. 354–365., Edinburgh, Scotland, UK.

[9] L. Sauvage, S. Guilley, J.-L. Danger, Y. Mathieu, and M. Nassar, "Successful Attack on an FPGA-based WDDL DES Cryptoprocessor Without Place and Route Constraints," in *DATE*. Nice, France: IEEE Computer Society, apr 2009, pp. 640–645.

[10] T. Popp and S. Mangard, "Masked Dual-Rail Pre-charge Logic: DPA-Resistance Without Routing Constraints," in *Proceedings of CHES'05*, ser. LNCS, vol. 3659. Springer, Sept 2005, pp. 172–186., Edinburgh, Scotland, UK.

[11] K. Tiri and I. Verbauwhede, "Place and Route for Secure Standard Cell Design," in *Proceedings of WCC / CARDIS*, Kluwer, Ed., Aug 2004, pp. 143–158, Toulouse, France.

[12] S. Guilley, P. Hoogvorst, Y. Mathieu, and R. Pacalet, "The "Backend Duplication" Method," in *CHES*, ser. LNCS, vol. 3659. Springer, 2005, pp. 383–397, August 29th – September 1st, Edinburgh, Scotland, UK.

[13] S. Guilley, S. Chaudhuri, L. Sauvage, T. Graba, J.-L. Danger, P. Hoogvorst, V.-N. Vong, and M. Nassar, "Place-and-Route Impact on the Security of DPL Designs in FPGAs," in *HOST (Hardware Oriented Security and Trust), IEEE*, Anaheim, CA, USA, jun 2008, pp. 29–35.

[14] Z. Chen and Y. Zhou, "Dual-Rail Random Switching Logic: A Countermeasure to Reduce Side Channel Leakage," in *CHES*, ser. LNCS, vol. 4249. Springer, 2006, pp. 242–254, Yokohama, Japan, http://dx.doi.org/10.1007/11894063_20.

[15] N. H. Weste and D. Harris, *CMOS VLSI Design: A Circuits and Systems Perspective*. Addison Wesley, 2004, 3 edition (May 11, 2004), ISBN: 0321149017.

[16] M. Saeki and D. Suzuki, "Security Evaluations of MRSL and DRSL Considering Signal Delays," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E91-A, no. 1, pp. 176–183, 2008, DOI: 10.1093/ietfec/e91-a.1.176.

[17] R. Soares, N. Calazans, V. Lomné, P. Maurine, L. Torres, and M. Robert, "Evaluating the robustness of secure triple track logic through prototyping," in *SBCCI'08: Proceedings of the 21st annual symposium on Integrated circuits and system design*. New York, NY, USA: ACM, 2008, pp. 193–198.

[18] J.-L. Danger and S. Guilley, "Circuit de cryptographie programmable – Logique BCDL (Balanced Cell-based Differential Logic)," 25 Mars 2008, Brevet Français FR08/51904, assigné à l'Institut TELECOM; WO/2009/118264.

[19] P. Yu and P. Schaumont, "Secure FPGA circuits using controlled placement and routing," in *CODES+ISSS'07: Proceedings of the 5th IEEE/ACM international conference on Hardware/software codesign and system synthesis*. New York, NY, USA: ACM, 2007, pp. 45–50.

[20] T. Akishita, M. Katagi, Y. Miyato, A. Mizuno, and K. Shibutani, "A Practical DPA Countermeasure with BDD Architecture," in *CARDIS*, ser. Lecture Notes in Computer Science, vol. 5189. Springer, Sept 2008, pp. 206–217, London, UK.

[21] K. Baddam and M. Zwolinski, "Divided Backend Duplication Methodology for Balanced Dual Rail Routing," in *CHES*, ser. LNCS, vol. 5154. Washington, DC, USA: Springer, aug 2008, pp. 396–410.

[22] R. P. McEvoy, C. C. Murphy, W. P. Marnane, and M. Tunstall, "Isolated WDDL: A Hiding Countermeasure for Differential Power Analysis on FPGAs," *ACM Trans. Reconfigurable Technol. Syst.*, vol. 2, no. 1, pp. 1–23, 2009.

[23] S. Bhasin, J.-L. Danger, F. Flament, T. Graba, S. Guilley, Y. Mathieu, M. Nassar, L. Sauvage, and N. Selmane, "Combined SCA and DFA Countermeasures Integrable in a FPGA Design Flow," in *ReConFig*. IEEE Computer Society, December 9–11 2009, p. 6 pages, Cancún, México, http://hal.archives-ouvertes.fr/hal-00411843/en/.

[24] K. Tiri, 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 *European Solid-State Circuits Conference (ESSCIRC)*, September 2002, pp. 403–406, Florence, Italy, http://citeseer.ist.psu.edu/tiri02dynamic.html.

[25] I. Verbauwhede and K. Tiri, "A Dynamic and Differential CMOS Logic with Signal-Independent Power Consumption to Withstand Differential Power Analysis," August 26 2008, uS patent 7,417,468, extended at world-level under reference WO/2005/029704, http://www.wipo.int/pctdb/en/wo.jsp?wo=2005029704.

[26] J. M. Rabaey, A. Chandrakasan, and B. Nikolic, *Digital Integrated Circuits*. Prentice Hall, 2003, iSBN-10: 0130909963, 761 pages.

[27] M. Bucci, L. Giancane, R. Luzzi, and A. Trifiletti, "Three-Phase Dual-Rail Pre-charge Logic," in *CHES*, ser. LNCS, vol. 4249. Springer, 2006, pp. 232–241, Yokohama, Japan.

[28] S. Guilley, P. Hoogvorst, Y. Mathieu, R. Pacalet, and J. Provost, "CMOS Structures Suitable for Secured Hardware," in *DATE'04 – Volume 2*. IEEE Computer Society, February 2004, pp. 1414–1415, paris, France.

[29] S. Guilley, F. Flament, R. Pacalet, P. Hoogvorst, and Y. Mathieu, "Secured CAD Back-End Flow for Power-Analysis Resistant Cryptoprocessors," *IEEE Design & Test of Computers*, vol. 24, no. 6, pp. 546–555, November-December 2007.

[30] ——, "Security Evaluation of a Balanced Quasi-Delay Insensitive Library," in *DCIS*. Grenoble, France: IEEE, nov 2008, 6 pages, Session 5D – Reliable and Secure Architectures, ISBN: 978-2-84813-124-5, full text in HAL: http://hal.archives-ouvertes.fr/hal-00283405/en/.

[31] S. Guilley, S. Chaudhuri, L. Sauvage, P. Hoogvorst, R. Pacalet, and G. M. Bertoni, "Security Evaluation of WDDL and SecLib Countermeasures against Power Attacks," *IEEE Transactions on Computers*, vol. 57, no. 11, pp. 1482–1497, nov 2008.

[32] L. Duflot, P. Le Moigne, and F. Germain, "Device Forming a Logic Gate for Minimizing the Differences in Electrical or Electromagnetic Behavior in an Integrated Circuit Manipulating a Secret," November 9 2006, Patent from the État Francais, représenté par le secrétariat général de la défense nationale, WO/2006/117391, http://www.wipo.int/pctdb/en/wo.jsp?WO=2006117391.

[33] F. Germain, "Towards cryptographic security using dedicated integrated circuits design methodologies," Ph.D. dissertation, École Polytechnique, Palaiseau, France, June 2006, http://www.imprimerie.polytechnique.fr/Theses/Files/Germain.pdf (french).

[34] M. Allam and M. Elmasry, "Dynamic current mode logic (DyCML), a new low-power/high-performance logic family," in *CICC*, 2000, pp. 421–424, dOI: 10.1109/CICC.2000.852699.

[35] F. Macé, F.-X. Standaert, J.-J. Quisquater, and J.-D. Legat, "A Design Methodology for Secured ICs Using Dynamic Current Mode Logic," in *PATMOS*, ser. Lecture Notes in Computer Science, vol. 3728. Springer, September 21–23 2005, pp. 550–560, Leuven, Belgium.

[36] F. Regazzoni, S. Badel, T. Eisenbarth, J. Großschädl, A. Poschmann, Z. Toprak, M. Macchetti, L. Pozzi, C. Paar, Y. Leblebici, and P. Ienne, "A Simulation-Based Methodology for Evaluating DPA-Resistance of Cryptographic Functional Units with Application to CMOS and MCML Technologies," in *International Conference on Embedded Computer Systems: Architectures, Modeling, and Simulation (SAMOS IC 07)*, July 2007, Samos, Greece.

[37] F. Regazzoni, T. Eisenbarth, A. Poschmann, J. Großschädl, F. K. Gürkaynak, M. Macchetti, Z. T. Deniz, L. Pozzi, C. Paar, Y. Leblebici, and P. Ienne, "Evaluating Resistance of MCML Technology to Power Analysis Attacks Using a Simulation-Based Methodology," *Transactions on Computational Science*, vol. 4, pp. 230–243, 2009.

[38] F. Regazzoni, A. Cevrero, F.-X. Standaert, S. Badel, T. Kluter, P. Brisk, Y. Leblebici, and P. Ienne, "A Design Flow and Evaluation Framework for DPA-Resistant Instruction Set Extensions," in *CHES*, ser. Lecture Notes in Computer Science, vol. 5747. Springer, 6-9 September 2009, pp. 205–219.

[39] G. Bouesse, M. Renaudin, B. Robisson, E. Beigné, P.-Y. Liardet, S. Prevosto, and J. Sonzogni, "DPA on Quasi Delay Insensitive Asynchronous Circuits: Concrete Results," in *XIX Conference on Design of Circuits and Integrated Systems, Proceedings of DCIS'04)*, 24–26 Nov 2004, Bordeaux, France (PDF).

[40] G. F. Bouesse, M. Renaudin, S. Dumont, and F. Germain, "DPA on Quasi Delay Insensitive Asynchronous Circuits: Formalization and Improvement," in *Proceedings of DATE'05*. IEEE Computer Society, March 2005, pp. 424–429, Munich, Germany.

[41] I. Hassoune, F. Macé, D. Flandre, and J.-D. Legat, "Dynamic differential self-timed logic families for robust and low-power security ICs," *Integration, the VLSI Journal*, vol. 40, no. 3, pp. 355–364, April 2007, DOI: 10.1016/j.vlsi.2006.04.001.

[42] T. Toffoli, "Reversible Computing," in *Proceedings of the 7th Colloquium on Automata, Languages and Programming*. London, UK: Springer-Verlag, 1980, pp. 632–644, London, UK.

[43] E. Fredkin and T. Toffoli, "Conservative Logic," *International Journal of Theoretical Physics*, vol. 21, no. 3/4, pp. 219–253, 1982, DOI: 10.1007/BF01857727.

[44] P. Kerntopf, "A new heuristic algorithm for reversible logic synthesis," in *DAC'04: Proceedings of the 41st annual Design Automation Conference*. New York, NY, USA: ACM, June 7–11 2004, pp. 834–837.

[45] R. Wille and D. Große, "Fast exact Toffoli network synthesis of reversible logic," in *ICCAD'07: Proceedings of the 2007 IEEE/ACM international conference on Computer-aided design*. Piscataway, NJ, USA: IEEE Press, 2007, pp. 60–64, DOI: 10.1109/ICCAD.2007.4397244.

[46] A. D. Vos and Y. Rentergem, "Reversible Computing: from Mathematical Group Theory to Electronical Circuit Experiment," in *CF (Computing Frontiers)*. ACM, May 4-6 2005.