

## Evaluating the Robustness of Secure Triple Track Logic Through Prototyping

Rafael Soares, Ney Calazans, Victor Lomné, Philippe Maurine, Lionel Torres, Michel Robert

► **To cite this version:**

Rafael Soares, Ney Calazans, Victor Lomné, Philippe Maurine, Lionel Torres, et al.. Evaluating the Robustness of Secure Triple Track Logic Through Prototyping. SBCCI'08: Symposium on Integrated Circuits and Systems Design, Sep 2008, Gramado, Brazil, ACM, pp.193-198, 2008, <<http://www.inf.ufrgs.br/chipinthepampa2008/>>. <10.1145/1404371.1404425>. <lirmm-00373516>

**HAL Id: lirmm-00373516**

**<https://hal-lirmm.ccsd.cnrs.fr/lirmm-00373516>**

Submitted on 6 Apr 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.

# Evaluating the Robustness of Secure Triple Track Logic through prototyping

Rafael Soares, Ney Calazans  
Pontifícia Universidade Católica do Rio Grande do Sul  
Faculdade de Informática - FACIN - PUCRS  
Av. Ipiranga, 6681 - 90619-900 Porto Alegre - Brazil

{rsoares,calazans}@inf.pucrs.br

## ABSTRACT

Side channel attacks are known to be efficient techniques to retrieve secret data. Within this context, this paper proposes to prototype a logic called Secure Triple Track Logic (STTL) on FPGA and evaluate its robustness against power analyses. More precisely, the paper aims at demonstrating that the basic concepts on which this logic leans are valid and may provide interesting design guidelines to obtain secure circuits.

## Categories and Subject Descriptors

B.6.[Hardware]: Logic Design

## General Terms

Security

## Keywords

DPA, CPA, Side-Channel-Attacks, DES, FPGA, Logic Style.

## 1. INTRODUCTION

In the last century, modern cryptology has focused mostly in defining cryptosystems that resist to logical attacks. In the last few years, the increasing use of secure embedded systems led researchers to focus also on the correlation between the data processed by cryptographic devices and their physical information leakage. As a result, efficient side-channel attacks appeared. These employ the device power consumption to disclose its secret key. Examples of such attacks are Simple Power Analysis (SPA) and Differential Power Analysis (DPA) [1].

Several countermeasures to these attacks have been proposed in previous works [2-4,11,14,16]. Most of these aim at hiding or masking the correlation between processed data and physical leakages. Adding random power consumption is one example technique employed.

Self-timed circuits seem an interesting implementation alternative, since it is more difficult to correlate the leaking syndromes to the data flow in a secure design in the absence of a clock signal [4,8].

Among all asynchronous circuit families, Quasi-Delay Insensitive (QDI) circuits offer another main advantage, the return to zero dual rail encoding used to present logic values [5,12]. Indeed, a rising transition on one of the two wires indicates that a bit is set to an invalid value with no logical meaning. Consequently, the transmission of a valid logic '1' or '0' always requires switching a rail to  $V_{DD}$ . Therefore, the differential power

Victor Lomné, Philippe Maurine,  
Lionel Torres, Michel Robert  
LIRMM, UMR 5506, Univ. Montpellier 2, CNRS  
161, rue Ada, 34392 Montpellier, France

{lomne,pmaurine,torres,robert}@lirmm.fr

signature of QDI circuits may be strongly reduced, given the use of symmetric cells.

Several implementations of robust dual rail cells are available in literature [10-15]. Most of these have been proposed to design robust ASIC, even if some works choose to map secure dual rail logic on FPGAs [6,7].

An investigation of the effective robustness against DPA of dual rail logic has been introduced in [16]. This evaluation demonstrates that the load imbalances (and thus power and timing ones) introduced during place and route steps significantly reduce the dual rail logic robustness against DPA. More precisely, the authors of [16] identify the potential asymmetrical propagation of data through the design as the remaining Achilles heel of dual rail logic. As a result, these authors introduced an improved dual rail logic, called Secure Triple Track Logic. The main characteristics of this logic are: a quasi-data independent power consumption and a quasi-data independent computation time at block level.

This paper investigates the efficiency of the concepts enclosed in the design guidelines of the Secure Triple Track Logic. After implementing and prototyping the most sensitive block of the DES algorithm on FPGA using STTL concepts, it proposes to evaluate the prototype robustness against DPA and Correlation Power Analysis (CPA).

The remainder of this paper is organized as follows. Section 2 briefly presents the STTL logic. Section 3 introduces the use of hard macros to efficiently map STTL on an FPGA. Here the quality of the obtained mapping is also discussed. Section 4 introduces the DPA and CPA platform used to evaluate the validity of the STTL concepts. Finally, Section 5 draws some conclusions.

## 2. SECURE TRIPLE TRACK LOGIC

Dual rail logic has been identified as an interesting countermeasure against DPA in several works [10-15]. This happens because its associated dual rail encoding theoretically allows reducing the correlation between its processed data and its power consumption. However, this claim holds if and only if some conditions are fulfilled [16].

As highlighted in [16], these conditions relate to the impact of the placement and routing steps on both the switching currents and the timing of dual rail designs. Indeed, place and route may introduce undesirable parasitic capacitances, be it in ASIC or programmable logic devices, unbalancing both the timing and the switching current profiles of dual rail gates and blocks. Place and

route are thus extremely critical steps in the design flow of secure dual rail circuits.

The STTL logic has been introduced to eliminate this weakness of dual rail against DPA and CPA. The main characteristics of this improved dual rail logic are quasi-data independent computation time and power consumption. These characteristics are obtained thanks to the introduction of a third rail as shown Fig. 1. The additional output wire, named  $S^V$ , indicates whenever the output data is stable and valid or not. Similarly, there are two additional input wires  $A^V$  and  $B^V$ , which indicate the validity of the incoming signals A and B.

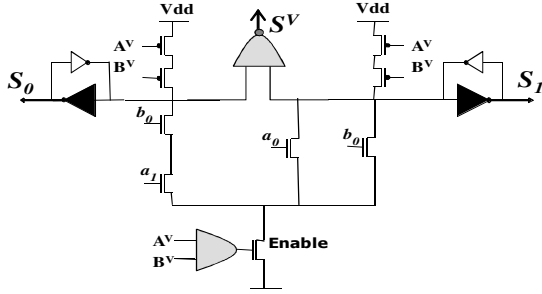


Fig. 1 STTL And2 gate.

STTL therefore operates according to a triple rail data encoding as represented in Fig. 2. This is not the first time that the use of an additional wire to encode the validity of a signal is proposed. This was first proposed in [9]. However, the goal of the additional wire in [9] is to obtain an “efficient hardware implementation” and not improve the design security or providing data independent logic.

As Fig. 2 illustrates, STTL is not a true triple rail encoding since the additional code value does not convey any information about the bit value, and is therefore not correlated with the value of this bit. This property is extremely important from a security point of view. Indeed, one key design characteristic of STTL cells is that all validity signals ( $A^V$ ,  $B^V$  and  $S^V$ ) are delivered by low switching current gates (greyed gates in Fig. 3), i.e. gates having greater propagation delays than high switching current gates (blackened gates in Fig. 3) in order to ensure that all input validity signals ( $A^V$ ,  $B^V$ ) settle after the data signals ( $A_0$ ,  $A_1$ ,  $B_0$ ,  $B_1$ ). This property can be easily obtained in ASIC design, by sizing transistors of greyed cells to be smaller than those of blackened cells.

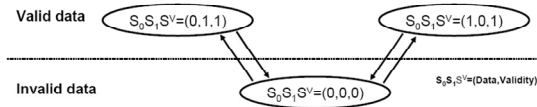


Fig. 2 Data encoding used in STTL.

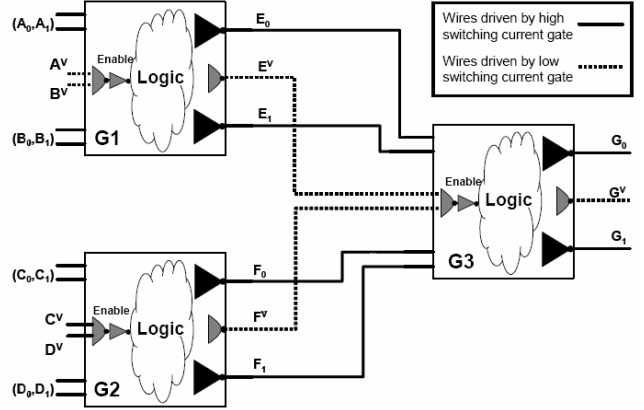


Fig. 3 Illustration of the basic operation of STTL.

The use of the additional wire implies, at cell level, a data independent power overhead estimated roughly to be within 10% to 30% compared to dual rail, as demonstrated in [10], depending on the complexity of the cell. However, this allows designing cells that have two interesting properties from a security point of view:

- A quasi data independent power consumption, as most of the proposed secure dual rail logic styles provide;
- A quasi data independent propagation delay, since the switching of the cell will always be triggered by a data independent signal (Enable) computed from validity signals ( $A^V$ ,  $B^V$  and  $S^V$ ) which are also data independent.

The second property above compensates the identified weakness of basic or secure dual rail cell introduced in former works, i.e. the arrival time imbalance, also called *early propagation* in [17]. In order to illustrate how the use of STTL cell warrants obtaining a quasi data independent computation time at circuit level, consider the simple circuit structure depicted in Fig. 3. This circuit is composed by three STTL cells. Once again, greyed and blackened elements correspond respectively to low and high switching current gates. Without loss of generality, assume that low and high switching current cells have respectively a propagation delay equal to  $\theta^S$  (low current logic delay) and  $\theta^F$  (high current logic delay) with  $\theta^S/\theta^F > 1$ . With such an assumption Fig. 4 represents the sequence of events occurring in the structure of Fig. 3 considering that  $A^V$ ,  $B^V$ ,  $C^V$  and finally  $D^V$  all rise at  $t = 0$ . On this Figure, the greyed time windows  $U_{AT}$  represent the incertitude on the arrival time values of signals  $E_0$ ,  $E_1$ ,  $F_0$ ,  $F_1$ ,  $G_1$  and  $G_0$ , due to the differential load imbalance introduced in the place and route steps. Despite this incertitude on the arrival time values of  $E_0E_1$  and  $F_0F_1$ , it can be noted that  $G_3$  fires at time  $T = 3\theta^S + \theta^F$ .

The important point here is that T does not depend on the input data, provided that  $U_{AT} < Q_{AT} = \theta^S + \theta^F$ , thanks to the stage to stage synchronization. More precisely, in an STTL circuit, gate

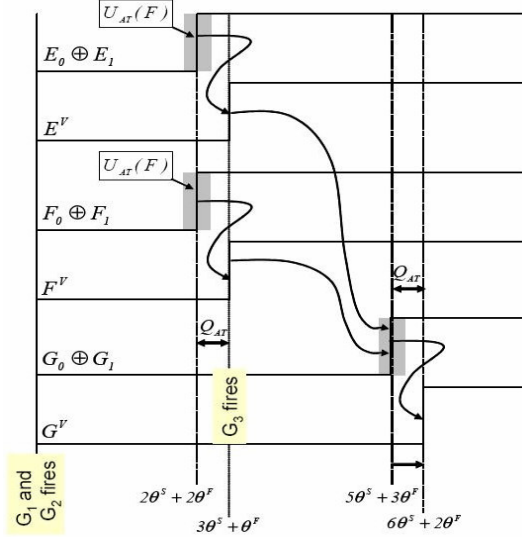


Fig. 4 Timing diagram associated to Fig. 3.

firing times are independent of the data processed, if the uncertainty on arrival time values, introduced during the place and route steps, is lower than  $Q_{AT} = \theta^S + \theta^F$ . The  $Q_{AT}$  time window, which is proportional to  $\theta^S$ , can be tuned by adequately sizing the delay  $Q_{AT}$ .

Thus, the robustness of an STTL circuit can be easily tuned through the adjustment of the width of this time window. While designing an ASIC, this can be achieved by properly sizing the gate and/or using dual  $V_T$  cells.

### 3. IMPLEMENTATION OF STTL ON FPGA

If many ASIC design techniques are available to tune the time window  $Q_{AT}$ , this task is more difficult while mapping on a field programmable device, since the hardware is fixed. This Section introduces a solution to map the concepts of STTL on FPGAs.

The first step to map STTL logic on FPGA is to design hard macros implementing basic STTL cells such as the STTL And2 function represented Fig. 1. A possible solution (certainly a sub-optimal one) to realize an STTL And2 function on FPGA is to integrate in a hard macro a functionality equivalent to an ASIC, as represented in Fig. 5a. In this Figure the logic is composed by C-Elements to avoid hazards on the inputs of traditional OR logic gates. Note that realizing this macro, the true and false data paths must be designed to have the same logical depth. This feature is important to obtain quasi independent power consumption and computation time at the cell level delays. To realize this on an FPGA, the solution is to implement independent logic. More specifically, the delay D of Fig. 5a has been obtained by cascading five LUTs. This allowed implementing a quasi independent timing logic for the validity signal having a constant and greater propagation delay than propagation delays of the true and false data paths, respectively.

Following these design guidelines, the mapping of an STTL And2 can be achieved using 11 LUTs (6 slices) as shown on Fig. 5b: 6 LUTs for the logic and 5 LUTs for the validation logic. Note that the area required to map a simple STTL cell on FPGA seems

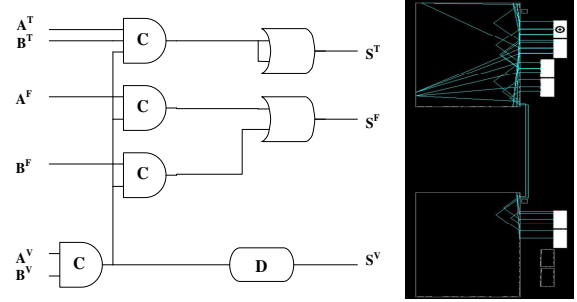


Fig. 5 (a) Logically and physically equivalent mapping of a STTL And2 gate and (b) picture of the obtained hard macro (Xilinx Spartan 3)

expensive. However it is necessary to keep in mind that the objective is to evaluate the validity of STTL concepts and not to find the best mapping of STTL on FPGAs. STTL cells mapping on LUTs could be improved. Similar results were obtained for other basic STTL cells.

## 4. EXPERIMENTATION

In order to evaluate the robustness of STTL against DPA, the most sensitive sub-module of a cryptographic algorithm has been implemented. The Data Encryption Standard (DES) was chosen because of it is a well known symmetric cryptosystem referenced by most studies on side-channel attacks. Only the critical module of the DES Cipher Function, called SBOX has been implemented.

### 4.1 DES sub-module characteristic

The chosen sub-module is depicted in Fig. 6. It takes the first 6 bits from the 48-bit expansion permutation output (Plaintext) and the first 6 bits output by the secret key compression permutation of DES first round (Secret sub key). These blocks are XORed bit-by-bit and the resulting 6-bit block is submitted to the Sbox1, which produces a 4-bit block as output. This module is sufficient to apply DPA.

Both single rail (SR) and STTL versions of this algorithm are available. The sub-module was implemented in single rail logic to validate the DPA/CPA flow, but also to obtain a reliable reference while evaluating the robustness against DPA/CPA of the STTL prototype. Table 1 gives the area required to implement the SR and STTL sub-modules on an FPGA. Table 1 also gives the results of timing analysis considering all possible input transitions and the 64 ( $2^6$ ) possible values of the Secret subkey part.

The obtained results demonstrate that the computation time of the STTL sub-module is, as expected, rigorously constant. However, the computation time is roughly five times greater than

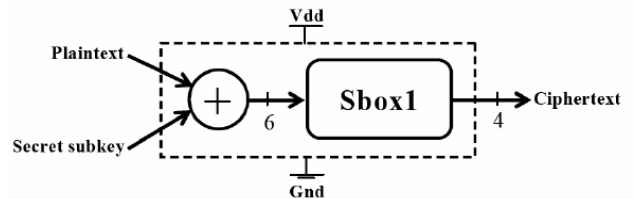


Fig. 6 Sub-module of DES Cipher function.

**Table 1. Sub-module area and timings**

	Single Rail	STTL
Min (ns)	15.627	102.664
Max (ns)	26.603	102.664
Average (ns)	22.231	102.664
Variation (ns)	10.976	0
Area (slices)	175	994
Area (%)	9%	51%

that obtained for the SR logic. The independent validation logic implementing delay D (Fig. 5a) on FPGA explains this result. However, this raises several questions not tackled in this paper for lack of space:

- Is a smaller D value sufficient to achieve constant computation time?
- Is there a way to implement this delay at a lower cost on current FPGAs?
- What would be the area of an FPGA dedicated to STTL design?

## 4.2 Measurement setup

To validate the STTL implementation, i.e. evaluate its robustness against DPA/CPA the following measurement setup, displayed in Fig 7, was employed:

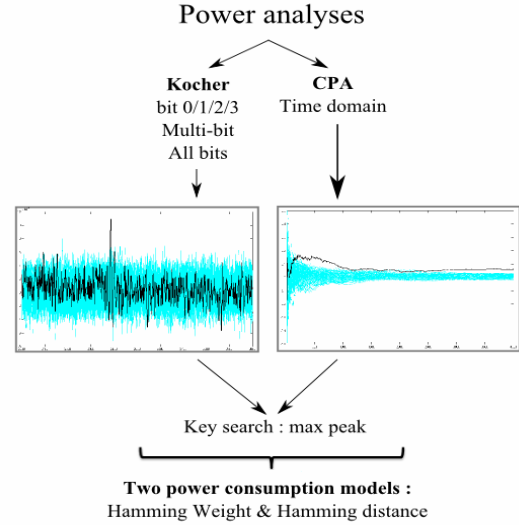
- A Xilinx Spartan3 board, the core voltage regulator of which has been disconnected, to supply the FPGA core using a less noisy battery;
- A current probe with a bandwidth of 1GHz, to measure the instantaneous switching current of the FPGA core;
- An oscilloscope to sample the switching current at 4GS/s;
- A PC to control the whole measurement setup, It provides data to the sub-module through an on chip RS232 module and stores measured power traces.

**Fig. 7 DPA/CPA Measurement setup.**

## 4.3 Performed DPA and CPA

In order to perform power analyses, power curves were collected on both single rail and STTL mappings. There are 64 power curves for the STTL (one for each possible data transition from the spacer value to a valid 6-bit value, see Fig. 2) and 4033 for the SR mapping (from any possible value to any others distinct value).

To reduce the noise and increase the Signal to Noise Ratio, each transition was applied 50 times to obtain, for each ciphering, an averaged power trace. Once data collection is done, power analyses were ran, based on two different power consumption

**Fig. 8 Overview of the DPA/CPA flow.**

models: the Hamming-Weight (HW) and Hamming-Distance (HD) models, illustrated in Fig. 8. Note:

- using the HW model resumes to count the number of logical '1' in the word targeted by the attack,
- while using the HD model resumes to count the number of bits that have switched from '0' ('1') to '1'('0') during a computation if the power curves are obtained by monitoring the supply (ground) rail.

We first performed some differential power analyses considering different selection functions. For these attacks, we used the selection function introduced by Kocher [1]. Four different analyses were undertaken, each targeting one output bit of the Sbox1. Next, we performed multi-bit differential analyses; i.e., we sorted the power traces according to the value of 2 output bits rather than 1. All power traces forcing respectively those two bits to the value '11' and '00' were thus gathered in the sets of power traces V1 and V0. Meanwhile all others power traces were discarded.

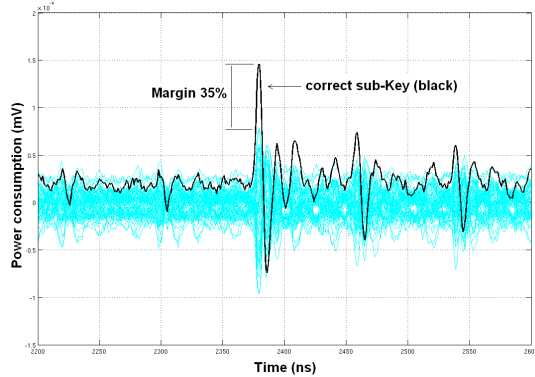
We then used two variants of Kocher selection function. These variants consist in considering respectively the Hamming Weight or the Hamming Distance of the four output bits of the Sbox1. Specifically, we defined two sets of power traces according to the value of the HW or HD, rather than to the value of one output bit.

Finally, we performed Correlation Power Analyses based on the HW and on the HD respectively. These analyses were performed in the time domain, i.e. one correlation value was computed for each sample of the power traces (between the instantaneous value of the current and either the HD or HW).

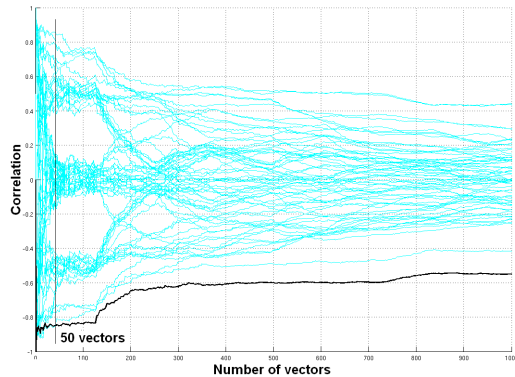
As illustrated in Fig. 9 and Fig. 10, all the above power analysis provided, in our case, 64 evolutions (one for each possible guess) of a quantity (a difference of current or correlation) versus time. Usually, the secret key corresponds (theoretically) to the guess resulting to the curve having the greatest amplitude.

## 5. RESULTS AND ANALYSIS

Even if theoretically, the guess corresponding to the secret key is characterized by the highest amplitude, in practice a margin should be considered, to warrant a high level of confidence while concluding about the successfulness of DPA or CPA.



**Fig. 9** Differential Power Analysis traces obtained for the SR DES sub-module (sub-key 10).



**Fig. 10** Correlation Power Analysis traces obtained for SR DES sub-module (sub-key 10).

Note that we defined this margin as the minimal relative difference between the amplitude of the DPA trace obtained for the right key, and the amplitude obtained for wrong guesses of the keys. We considered that a DPA or a CPA was successful if the resulting margin is larger than 10%.

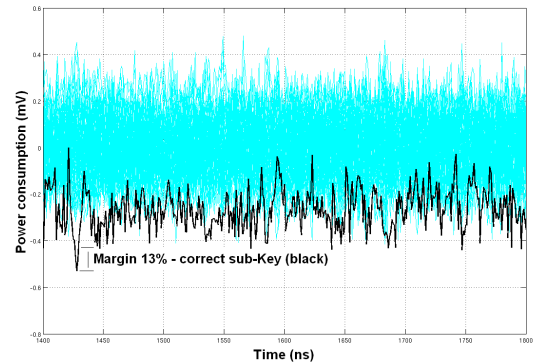
All the power analyses described in the preceding Section were first applied on the single rail DES sub-module. The analyses were done using an input sequence of 4033 different vectors. This sequence was defined to obtain the average power traces of all possible input transitions (6 bits). For each considered sub-key value, most differential power analyses were successful, since the margins obtained range between 10% and 30%, depending on the considered selection function. Moreover, during these analyses, we observed that the Hamming Distance model gives, as expected, higher margins than the Hamming Weight model. As an illustration, Fig. 9 gives the differential power analysis traces obtained for the sub-key 10, while Fig. 10 represents the evolution of the correlation coefficients with respect to the number of input vectors used to perform the correlation power analysis. As shown on the latter figure, 50 inputs are usually sufficient to reveal the secret sub-key, even if the statistical convergence is not reached.

In a second experiment, we applied all power analyses described Section 4, on the STTL DES sub-module. These analyses were performed for all possible values of the secret key. Table 2 summarizes the obtained results. As shown, only the secret key 57 was revealed with a high level of confidence by only one power analysis (a DPA based on the Hamming Weight model). The margin was indeed equal to 13% (>10%). However, as shown, Fig. 11 this margin is obtained on an extremely short time interval and maybe this peak is a ghost one.

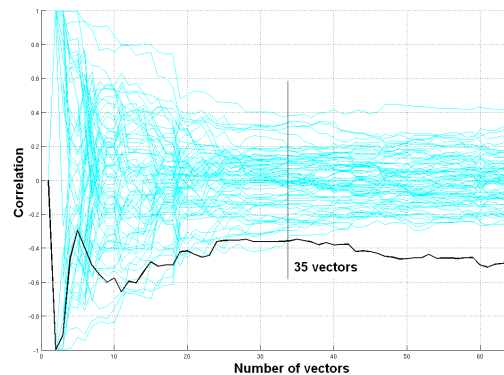
**Table 2** Power analyses results on STTL.

Sub-key	HW model	HD model
0 → 34	power analyses failed	power analyses failed
35	power analyses failed	success (DPA Kocher bits 0 & 2) margin : 8%
36 → 56	power analyses failed	power analyses failed
57	Success (DPA Kocher bit 0) Margin : 13%	power analyses failed
58 → 63	power analyses failed	power analyses failed

These results demonstrate the robustness of the STTL DES sub-module. It demonstrates that designing quasi-data independent computation time and power consumption module is a good solution to increase the robustness against DPA and CPA.



**Fig. 11** Differential Power Analysis traces obtained for the STTL DES sub-module (sub-key 57)



**Fig. 12** Correlation Power Analysis traces obtained for STTL DES sub-module (sub-key 57)

## 6. CONCLUSIONS

In this paper, we prototyped on FPGA a logic called Secure Triple Track Logic and evaluated its robustness against power analyses. We demonstrated that obtaining *simultaneously* quasi-data independent power consumption and computation time constitutes an interesting design guideline to increase the robustness of circuits against differential and correlation power analyses.

## 7. ACKNOWLEDGMENTS

Our thanks to: the CAPES-COFECUB, the 'Pôle de Competitivité SCS' and the ANR for grants and supports.

## 8. REFERENCES

- [1] P. Kocher, J. Jaffe and B. Jun, "Differential Power Analysis," in *Proc. 19th International Conference on Cryptology (CRYPTO)*, pp. 388–397, Aug. 1999.
- [2] Z. Chen and Y. Zhou, "Dual-Rail Random Switching Logic: A Countermeasure to Reduce Side Channel Leakage," in *Proc. 8th Workshop on Cryptographic Hardware and Embedded Systems (CHES)*, pp. 242-254, Oct. 2006.
- [3] A. Bystrov, A. Yakovlev, D. Sokolov and J. Murphy, "Design and Analysis of Dual Rail Circuits for security Applications," *IEEE Transactions on Computers*, vol. 54, no. 4, pp. 449-460, Apr. 2005.
- [4] J. J. A. Fournier, S. W. Moore, H. Li, R. D. Mullins and G. S. Taylor, "Security Evaluation of Asynchronous Circuits," in *Proc. 5th Workshop on Cryptographic Hardware and Embedded Systems (CHES)*, pp. 137-151, Sept. 2003.
- [5] G. F. Bouesse, M. Renaudin, S. Dumont, F. Germain, "DPA on Quasi Delay Insensitive Asynchronous Circuits : Formalization and Improvement," in *Proc. Design, Automation and Test in Europe Conference and Exposition (DATE)*, pp. 424-429, Mar. 2005.
- [6] K. Tiri and I. Verbauwhede, "A Logic Level Design Methodology for a Secure DPA Resistant ASIC or FPGA Implementation", in *Proc. Design, Automation and Test in Europe Conference and Exposition (DATE)*, pp. 246-251, Feb. 2004.
- [7] F. X. Standaert, S. B. Ors and B. Preneel, "Power Analysis of an FPGA: Implementation of Rijndael: Is Pipelining a DPA Countermeasure?", in *Proc. 6th Workshop on Cryptographic Hardware and Embedded Systems (CHES)*, pp. 30-44, Aug. 2004.
- [8] Z.- C. Yu, S. B. Furber and L. A. Plana, "An Investigation into the Security of Self-Timed Circuits," in *Proc. 9th International Symposium on Asynchronous Circuits and Systems (ASYNC)*, pp. 206–215, May. 2003.
- [9] T. H. Y. Meng, R. W. Brodersen and D. G. Messerschmitt, "Automatic Synthesis of Asynchronous Circuits from High-Level Specifications," *IEEE Transaction on Computer Aided Design*, vol. 8, no. 11, pp. 1185–1205, Nov. 1989.
- [10] A. Razafindraibe, M. Robert, P. Maurine "Improvement of dual rail logic as a countermeasure against DPA", *IFIP International Conference on Very Large Scale Integration, 2007. VLSI-SoC 2007*, pp. 270-275, Oct. 2007.
- [11] S. Guilley, P. Hoogvorst, Y. Mathieu, R. Pacalet and J. Provost, "CMOS Structures Suitable for Secure Hardware," in *Proc. Design, Automation and Test in Europe Conference and Exposition (DATE)*, pp. 1414–1415, Feb. 2004.
- [12] A. Razafindraibe, P. Maurine, M. Robert, F. Bouesse, Bertrand Folco and M. Renaudin, "Secured Structures for Secured Asynchronous QDI Circuits," in *Proc. 19th International Conference on Design of Circuits and Integrated Systems (DCIS)*, pp. 20-26, Nov. 2004.
- [13] K. Tiri and I. Verbauwhede, "Securing Encryption Algorithms against DPA at the Logic level: Next Generation Smart Cards Technology," in *Proc. 5th Workshop on Cryptographic Hardware and Embedded Systems (CHES)*, pp. 125–136, Sept. 2003.
- [14] F. Mace, F. Standaert, I. Hassoune, J.-D. Legat and J.-J. Quisquater, "A Dynamic Current Mode Logic to Counteract Power Analysis Attacks," in *Proc. 19th International Conference on Design of Circuits and Integrated Systems (DCIS)*, Nov. 2004.
- [15] K. J. Kulikowski, M. Su, A. B. Smirnov, A. Taubin, M. G. Karpovsky and D. MacDonald, "Delay Insensitive Encoding and Power Analysis: A Balancing Act," in *Proc. 11th IEEE International Symposium on Asynchronous Circuits and Systems (ASYNC)*, pp. 116–125, Mar. 2005.
- [16] A. Razafindraibe, M. Robert and P. Maurine, "Formal Evaluation of the Robustness of Dual-Rail Logic against DPA Attacks," in *Proc. 16th International Workshop on Power and Timing Modeling, Optimization and Simulation (PATMOS)*, pp. 634–644, Sept. 2006.
- [17] K.J. Kulikowski, V. Venkataraman, Z. Wang and A. Taubin, "Power Balanced Gates Insensitive to Routing Capacitance Mismatch," in *Proc. Design, Automation and Test in Europe Conference and Exposition (DATE)*, pp.1280-1286, Mar 2008.