

# Side-Channel Attack Mitigation Using Dual-Spacer Dual-Rail Delay-Insensitive Logic (D<sup>3</sup>L)

Washington Cilio<sup>1</sup>, Michael Linder<sup>1</sup>, Christopher Porter<sup>1</sup>, Jia Di<sup>1</sup>, Scott Smith<sup>2</sup>, and Dale Thompson<sup>1</sup>

<sup>1</sup>Department of Computer Science Computer Engineering, <sup>2</sup>Department of Electrical Engineering, University of Arkansas  
wcilio@uark.edu, mlinder@uark.edu, cporter@uark.edu, jdi@uark.edu, smithsco@uark.edu, drt@uark.edu

**Abstract**— Side-channel attacks have become a threat to secure electronic circuits, due to the strong correlation between data pattern and leaking power/timing information. By monitoring the power/timing behavior of a synchronous circuit, an attacker can easily obtain the secret data stored in the device. Although dual-rail asynchronous circuits have more stable power traces, they still show power fluctuation because of the imbalanced load between two rails. Moreover, asynchronous circuits are the most prone to timing attacks since delay is data dependent. Dual-spacer Dual-rail Delay-insensitive Logic (D<sup>3</sup>L), presented in this paper, is able to mitigate power and timing based side-channel attacks. Power fluctuation is decoupled from data pattern by the use of a dual-spacer protocol, while timing-data correlation is broken by insertion of random delays.

## I. INTRODUCTION

As technology advances, our electronics contain more and more personal information such as: bank accounts, identification numbers, passwords, and other sensitive data that need to be secured from unauthorized access. Although originally considered safe and secure, hardware, just as software, is prone to attacks that force the targeted system to reveal such data. Hardware attacks usually take place while data is being processed locally (e.g., computers, cell phones, GPS devices). Hence, cryptographic keys, secure information, and other important data may be in jeopardy if an attacker is able to perform a “hack” on that system.

Cryptographic algorithms are commonly used to protect sensitive data. However, despite of the mathematical robustness of these algorithms, their physical implementations are known to be susceptible to attacks that can jeopardize the secrecy of the information. Non-invasive attacks on such devices take advantage of side channel information that is leaked from the system instead of trying to reverse engineer it. Such information can be power, timing, electromagnetism, and any other information that might be measured from the device during computation.

Kocher et al. present in [1] the Simple Power Analysis (SPA) and Differential Power Analysis (DPA) for the Data Encryption Standard (DES) algorithm. SPA is a simple technique that directly correlates data being processed with the power consumption of the device. On the other hand, DPA

uses statistical analysis to correlate data and power/timing information. Similarly in [10], Correlation Power Analysis (CPA) uses the Pearson product-moment correlation coefficient to compare a key guess to the power/timing data.

Cryptosystem devices can also become vulnerable if timing information can be collected from the device. Kocher in [3] exploits this timing vulnerability by measuring the period of time it takes to perform operations that involve the cryptographic key. Such vulnerabilities can come from RAM cache hits/misses, branch instruction, conditional statements, and optimizations to the device.

Side channel countermeasures are critical to the safeguard of secret information inside cryptosystems. Both power and timing exploitation have to be mitigated in order to ensure a tamper-resistant design. Power mitigation techniques such as: noise introduction, signal sizes reduction, and aggressive shielding can be used to make the attack more difficult. Kocher in [1] suggests introducing noise into the power consumption measurements. However, noise can be reduced or eliminated with hardware/software noise filters, which can significantly shrink the number of samples needed to uncover information [4]. In the same fashion, reducing the signal sizes increases the number of samples needed to uncover a key. Nevertheless, it is still possible to obtain a key from power samples. On the other hand, aggressive shielding can make the attacks ineffective, but it drastically adds cost and size to the design [1].

Timing alternation techniques have also been developed to deter attacks. The most intuitive technique to mitigate timing attacks is to make all operations run in constant time [3]. This completely eliminates differences between operations and decouples the timing-data correlation. However, external factors such as RAM cache hits/misses, different execution time instructions, and compiler optimizations make constant execution times hard to achieve.

Asynchronous circuits, on the other hand, possess known characteristics that could help mitigate such attacks. Fant et al. in [5] present one of the most popular quasi delay-insensitive asynchronous architectures named NULL Convention Logic (NCL). NCL uses input completeness, multi-rail encoding, and threshold gates with hysteresis. These characteristics allow switching activity in a device to be independent from the input

and only determined by the number of data processed [2], making power variation significantly smaller than synchronous designs [6]. Nonetheless, switching activity remains unbalanced among the rails with different capacitive loads; thus, DPA, High-Order DPA, or CPA may still be able to succeed. The advantage, however, is that the amount of samples dramatically increases. Unfortunately, NCL shows a weakness due to a strong data-timing dependency, making the device susceptibility to timing attacks.

This paper presents the design, implementation, and simulation of a Dual-spacer Dual-rail Delay-insensitive Logic ( $D^3L$ ) that is able to mitigate power and timing based side-channel attacks. Some key design characteristics to be presented include: decoupling of power fluctuation from data pattern by using dual spacers, separating timing-data correlation by inserting random delays.

This paper will cover the topics in the following order: Section 2 explains the circuit architecture that makes  $D^3L$  suitable to mitigate side-channel attacks. It also introduces the Advanced Encryption Standard (AES) architecture, which is implemented using  $D^3L$ , Null Convention Logic (NCL), and synchronous logic to test and compare the effectiveness of the mitigation. Section 3 explains and confirms the operation of the  $D^3L$  AES design. The section shows how the analysis is performed, and the assumptions made for the results. It ends by presenting a comparison among  $D^3L$ , NCL, and synchronous design. Section 4 summarizes the work done in this paper.

## II. DESIGN & IMPLEMENTATION

NCL represents its data with three states: Data 0, Data 1, and NULL or spacer state as shown on Table I. These states need to be coded using two rails (wires). Each rail is mutually exclusive from the other one, meaning both rails cannot be asserted at the same time. While asserting a TRUE value on a rail represents Data 0, asserting the other represents Data 1. This single spacer scheme, however, has been proved to fall short in balancing the switching activity between two rails.

Due to the return-to-spacer protocol, NCL circuits, after a data cycle, must always return to the NULL state before accepting new data. While using such protocol decouples data from switching activity of the two-rail bundle, the unbalanced switching between rails still exists.  $D^3L$  solves unbalanced switching by adding a new spacer. In the same fashion as NCL,  $D^3L$  also represents its states by asserting one rail at a time to denote Data 0 or Data 1. While for NCL asserting a TRUE value in both rails results in an error state,  $D^3L$  takes advantage of this invalid state and uses it as a new spacer while keeping the old spacer, as shown in Table 1. The new state is known as all-one spacer, which denotes when both rails assert a TRUE value. At the same time, the NULL state takes the new name of all-zero spacer.

By alternating spacers in between data,  $D^3L$  creates a dual-spacer protocol that allows both rails to have identical

switching activity regardless of the data being processed. For example, Figure 1 shows a dual-spacer protocol sequence, which alternates from an all-zero spacer to an all-one spacer after every data set.

### A. $D^3L$ Gates

As an expansion of NCL,  $D^3L$  takes advantage of threshold gates. Furthermore,  $D^3L$  also provides the 27 basic gates, which are fundamental to embark all of the functions that can be created with 4 inputs or less. As shown on Figure 2, in the same manner as the threshold gate notation (TH $mn$ ) for NCL, every  $D^3L$  gate has an  $n$ -input,  $m$ -threshold, denoted by  $D^3Lmn$ . For example, the  $D^3L34$  gate has A, B, C, and D as its inputs, and will only assert its output when 3 or more of its inputs have been asserted. Its behavior is equivalent to the Boolean equation  $ABC+ABD+ACD+BCD$ .

Unlike NCL gates, which have hysteresis and hold their value until all inputs have been de-asserted,  $D^3L$  gates will not hold the output value once inputs are de-asserted below the threshold. The omission of hysteresis in  $D^3L$  gates accommodates the all-one spacer. However, input completeness is now compromised, and cannot be guaranteed anymore.

Input completeness requires the output of a function not to transition, from a spacer state to a data state and vice versa, before all inputs have made the same transition. Even though  $D^3L$  cannot guarantee input completeness on its own, NCL\_X technique, presented in [7], provides extra circuitry to preserve input completeness in the design. As shown in Figure 3, the inputs and outputs of the AND function, which is made up of a  $D^3L22$  and a  $D^3L34w22$  (the w22 means input A and B of the gate weight twice as much), are connected to XNOR gates and merged into one signal using a TH33 gate to ensure the completeness of the gates.



Figure 1. Dual-spacer protocol sequence.

TABLE I. NCL &  $D^3L$  TRUTH TABLE STATE

State Code		NCL	$D^3L$
Rail 1	Rail 0		
0	0	NULL spacer	All-zero spacer
0	1	Data 0	Data 0
1	0	Data 1	Data 1
1	1	Invalid	All-one spacer

### B. $D^3L$ Registers

A basic  $D^3L$  register has two basic components: a modified NCL register, and a KI generator. An NCL register contains three gates: two TH22 to save data and a TH12 to generate handshake signals. A modified NCL register, however, replaces the TH12, which only detects all-zero spacers, with a two-input XNOR gate to generate the same handshake signal

with the exception that all-one spacers can also be detected. On the other hand, a KI generator generates the correct signal to control the latching of data or spacer depending on the state of the register.

Additionally, D<sup>3</sup>L ring registers will enter deadlock if not properly handled. Deadlock is created due to the lack of dual-spacers inside the loop, because basic D<sup>3</sup>L registers do not have the capabilities to generate alternating spacers. A D<sup>3</sup>L filter register is used to supply the proper spacers inside ring registers. The spacer filter component, inside the filter register, analyzes the dual-rail input along with the previous spacer (ps) signal to properly alternate the spacer needed to continue the dual-spacer protocol.

### C. AES Design Using D<sup>3</sup>L

The Advanced Encryption Standard (AES) algorithm described in [8], can process input blocks, also known as plaintext, of 128 bits using cipher key lengths of 128, 192, and 256 bits, which iterate through the algorithm 10, 12, and 14 rounds, respectively. In every round, the State, which is an intermediate output seen as a 4×4 two-dimensional array, is subject to SubByte, ShiftRows, MixColumns, and AddRound transformations, except for the last round in which the MixColumns transformation is omitted. On the other hand, a different sub-key is created every round using the key expansion routine. The sub-key is then added to the State with an AddRound transformation. An AES D<sup>3</sup>L, NCL, and synchronous cores were designed to compare the resistance to side-channel mitigation.

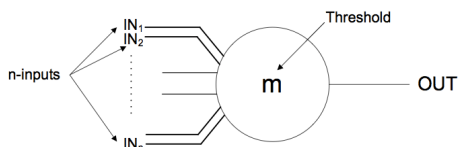


Figure 2. D3Lmn gate

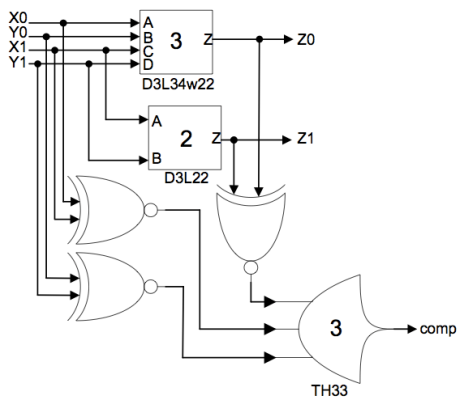


Figure 3. Input complete D<sup>3</sup>L AND function with completion logic.

## III. SIMULATION RESULTS

Since the focus of this work is to measure the resistance of the featured AES D<sup>3</sup>L design against side-channel attacks, simulations are performed at the sub-circuit level instead of a whole design. The main reason is simulation time. A Full design simulation takes extremely long time to simulate one sample at transistor level. However, The sub-circuit used for simulations can be found as part of an AES operation. The original 128-bit secret key and plaintext undergo an AddRound transformation. Additionally, a SubByte transformation, which contains 16 S-Boxes that take an input of 8 bits each, is then applied to each output byte of the AddRound. It is only necessary to attack one S-Box at a time, which allows 2<sup>8</sup> or 256 possible key combinations for each S-Box. Although brute force, the attack is actually very fast since there are only 256 combinations. Furthermore, it is assumed that if one byte is successfully discovered, the other 15 bytes can be hacked in a similar fashion.

CPA was used to attack all three designs for power, energy and timing attacks. Its strong statistical model makes the attack even more effective than the original DPA.

Because both asynchronous designs lack of a clock signal to guide the attacker, the switching activity can happen at different times, which creates small misalignments between side-channel signals. However, an attacker can eliminate the misalignments by using the side-channel energy. As shown in [9], the attacker can break the current fluctuation into smaller pieces and compute the energy for each. This approach is chosen over power to attack both asynchronous designs.

Additionally, timing attack is only performed against the D<sup>3</sup>L design for two reasons. First, it is well known that sequential synchronous designs are resistant to timing attack due to their fixed-time operation, in which timing is not dependent on the data processed. Therefore, an attack to a synchronous is unlikely to yield positive results. Second, while NCL as well as D<sup>3</sup>L have inherent timing-data dependencies, i.e., the timing is dependent on the data being processed; there is more interest in analyzing the timing-data masking performance on the D<sup>3</sup>L design than NCL. However, the technique used can also be applied to an NCL circuit. The active time of the computation is used as the time reference to analyze the timing-data correlation.

TABLE II. CPA RESULTS FOR D<sup>3</sup>L, NCL AND SYNCHRONOUS DESIGNS.

Design	Number of Patterns	Correct Key Guess Success/Fail	Maximum Correlation Coefficient	Data Type Collected
D <sup>3</sup> L	256	Failed	0.354	Energy
NCL	256	Success	0.428	Energy
Synchronous	256	Success	0.668	Power

TABLE III. TIMING ATTACK RESULTS FOR THE D<sup>3</sup>L DESIGN WITH AND WITHOUT DELAY USING CPA

Design	Number of Patterns	Correct Key Guess Success/Fail	Maximum Correlation Coefficient	Data Type Collected
D <sup>3</sup> L	256	Success	0.373	Active time
D <sup>3</sup> L delays	256	Fail	0.106	Active time

The results presented in Table II show that the attacks on the synchronous and NCL designs were successful. The success of the attack on the synchronous design was expected, because synchronous designs leak critical power information. Furthermore, the correlation coefficient ratifies with a high correlation rating that the synchronous design is very vulnerable to a power attack and can be broken easily. Similarly, the attack on the NCL design yield results that were also expected. Although the NCL design is more resistant to an attack, the unbalanced capacitances found on the dual-rails still leak side channel information that can be used. However, it is important to point out that the correlation coefficient is much lower than the synchronous design, and may yield better possibilities for an error on the key guess.

On the other hand, the energy attack against the D<sup>3</sup>L design presents a couple of reassuring results. First, the CPA program failed to guess the correct cipher key when using the energy data. And second, the program failed with the lowest correlation coefficient from all three attacks. That is, the CPA was not able to find a better correlation between the energy estimate and the actual data. Additionally, Table III shows the results for the timing attack on both the original D<sup>3</sup>L design, and the D<sup>3</sup>L design with delay elements. The success obtained for the timing attack on the original D<sup>3</sup>L was not a surprise, because strongly intertwined timing-data correlation exists in D<sup>3</sup>L and other asynchronous circuits. Thus, the timing attack was very likely to succeed on the original D<sup>3</sup>L design. Even though the correlation coefficient appears to be low (0.373), the timing-data correlation, at that level, is still strong enough to lead to the right key guess. However, the correlation coefficient was not expected to be that low, especially in a timing attack. On the other hand, the D<sup>3</sup>L design with added delay elements performed much better than the original. Not only the CPA could not guess the correct key, but also the correlation coefficient was very low (0.106), which leads one to believe there was very little correlation between the data and the estimate. Therefore, the delay elements successfully mask the timing dependencies in the design.

Conversely, implementing a D<sup>3</sup>L design incurs penalties in area and power. Two main sources for the penalties are input completeness logic, and the KI generators inside the registers. XNOR gates, which ensure functions are input-complete, adds expensive overhead in the form of area and power. Similarly, registers pay an extra area and power penalty due to the extra logic in the KI generator, which is needed to control the dual-spacer protocol.

## IV. CONCLUSION

This paper compared the effectiveness of the proposed D<sup>3</sup>L scheme against power and timing based side-channel attacks by using balanced switching rails as well as random delays. The results show that sensitive information can safely be processed using D<sup>3</sup>L while leaking almost non-correlated information to an attacker. Future work includes optimizing the D<sup>3</sup>L designs in reducing area/power overhead and completing the automated D<sup>3</sup>L IC design flow.

## REFERENCES

- [1] J. Jaffe, P. Kocher, and B. Jun, "Differential Power Analysis," in *Proc. 19th International Advances in Cryptology Conference CRYPTO '99*, Santa Barbara, CA, 2009, pp. 388-397, 1999
- [2] J. Di, F. Yang, "D<sup>3</sup>L – A Framework on Fighting Against Non-Invasive Attacks to Integrated Circuits for Security Applications," in *Proc. Third IASTED International Conference Circuits, Signals, and Systems*, Marina del Rey, CA, 2005, pp. 73-78.
- [3] P. C. Kocher, "Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS, and Other Systems," in *Proc. 16th International Advances in Cryptology Conference CRYPTO '96*, Santa Barbara, CA, 2006, pp. 388-397.
- [4] T. Messerges, E. Dabbish, and R. Sloan, "Examining Smart-Card Security under the Threat of Power Analysis Attacks," *J. IEEE Trans. Compt.*, vol. 51, no. 5, pp. 541-552, May 2002.
- [5] K. Fant and S. Brandt, "NULL Convention Logic<sup>TM</sup>: A Complete and Consistent Logic for Asynchronous Digital Circuit Synthesis," in *Proc. Application Specific Systems, Architectures and Processors ASAP '96*, 1996, Chicago, IL, pp 261-273.
- [6] J. Di, J. S. Yuan, and M. Hagedorn, "Energy-aware Multiplier Design in Multi-rail Encoding Logic," in *The IEEE 45th Midwest Symposium on Circuits and Systems MWSCAS-2002*, vol. 2, pp. II-294-II-297, Aug. 2002.
- [7] A. Kondratyev, and K. Lwin, "Design of Asynchronous Circuits Using Synchronous CAD Tools," *J. IEEE Des. Test*, vol. 19, no. 4, pp. 107-117, July 2002.
- [8] National Inst. of Standards and Technology, "Federal Information Processing Standard 197, The Advanced Encryption Standard (AES)," URL: <http://csrc.nist.gov/publications/fips/fips197/fips197.pdf>, 2001.
- [9] Le Thanh-Ha, J. Clediere, C. Serviere, and J. Lacoume, "Efficient Solution for Misalignment of Signal in Side Channel Analysis," in *Conference on Acoustics, Speech and Signal Processing ICASSP 2007*, Honolulu, HI, 2007, pp. II-257-II-260 vol. 2.
- [10] E. Brier, C. Clavier, and F. Olivier, "Correlation Power Analysis with a Leakage Model," in *Proc. Cryptographic Hardware and Embedded Systems CHES 2004*, Cambridge, MA, Aug. 2004, pp. 16-29.