Hindawi Publishing Corporation
International Journal of Reconfigurable Computing
Volume 2010, Article ID 375245, 12 pages
doi:10.1155/2010/375245

Research Article

Exploiting Dual-Output Programmable Blocks to Balance Secure Dual-Rail Logics

Laurent Sauvage, Maxime Nassar, Sylvain Guilley, Florent Flament, Jean-Luc Danger, and Yves Mathieu

Département COMELEC, Institut TELECOM/TELECOM ParisTech, CNRS LTCI (UMR 5141), 46 Rue Barrault, 75 634 Paris Cedex 13, France

Correspondence should be addressed to Laurent Sauvage, laurent.sauvage@telecom-paristech.fr and Sylvain Guilley, sylvain.guilley@telecom-paristech.fr

Received 2 March 2010; Accepted 5 October 2010

Academic Editor: Lionel Torres

FPGA design of side-channel analysis countermeasures using unmasked dual-rail with precharge logic appears to be a great challenge. Indeed, the robustness of such a solution relies on careful differential placement and routing whereas both FPGA layout and FPGA EDA tools are not developed for such purposes. However, assessing the security level which can be achieved with them is an important issue, as it is directly related to the suitability to use commercial FPGA instead of proprietary custom FPGA for this kind of protection. In this article, we experimentally gave evidence that differential placement and routing of an FPGA implementation can be done with a granularity fine enough to improve the security gain. However, so far, this gain turned out to be lower for FPGAs than for ASICs. The solutions demonstrated in this article exploit the dual-output of modern FPGAs to achieve a better balance of dual-rail interconnections. However, we expect that an in-depth analysis of routing resources power consumption could still help reduce the interconnect differential leakage.

1. Introduction

During the last decade, a considerable number of countermeasures have been proposed to protect cryptographic devices against Side Channel Analysis (SCA). They customarily fall in two categories. The first one, *masking* [1, 2], is very popular in the smart card community as it can be implemented at the algorithm level. Intermediate data processed by a cryptoprocessor are concealed by a value called mask, randomly chosen, which in turn makes power consumption random. In the second category, *hiding*, intermediate values remain the same as for unprotected implementations, but power consumption is made as constant as possible. According to the state-of-the-art attacks, masked implementation on a Field Programmable Gates Array (FPGA) could be broken with Higher-Order Differential Power Analysis (HODPA) using 12,000 power consumption traces [3] whereas 1,500,000 measurements are not sufficient to disclose the entire secret key of an Application Specific Integrated Circuit (ASIC) cryptoprocessor protected by Wave Dynamic Differential Logic (WDDL) [4], the most popular hiding countermeasure developed by Tiri and Verbauwhede. Proposals for merging both techniques with a view to compensate for weakness of masking against HODPA and for backend operations hardness of hiding have been reported in [5, 6], but this approach unfortunately remains vulnerable when the masking relies on one single bit of entropy [7].

Unmasked dual-rail with Precharge Logics (DPLs) in general seem thus to be a sound solution. As WDDL is based on a standard cell flow, it is the most suited for FPGA implementation. Guidelines for synthesis can be found in [8, 9]. We notice incidentally that those articles target 4 → 1 LuT-based FPGA technologies and thus do not take full advantage of the advanced features of modern FPGAs, such as ALM (Adaptative Logic Modules) configurable blocks and dual-output logic blocks. DPL robustness against SCA relies on perfectly matching differential routing, which appears to be incredibly hard to achieve for large FPGA designs because of the following.

(i) The only way to evaluate the imbalance between two differential paths is to compute the difference of path delays. But as delays provided by development tools are maximum values, not typical, unbalance evaluation is coarse.

(ii) Routing resources are limited and have thus to be properly shared between each component of the design. This may affect placement, which could no longer simply consist in placing differential components side by side.

(iii) Commercial off-the-shelf (COTS) FPGA layout has been designed to provide flexibility, not differential capability. As a consequence, the performance of DPL is expected to be less efficient than when embedded in ASIC, for which differential components can be placed as close as possible, decreasing the impact of intradie process variability.

To overcome these problems, Yu and Schaumont suggest the Double WDDL (DWDDL) [10] design strategy: from a first direct WDDL module, a complementary clone with identical routing is obtained by duplication, relocation, and logic modification. This way, leakages due to imbalances of each module are expected to compensate one with each other. Unfortunately, some registers of DWDDL never go to precharge value and introduce leakages in the Hamming Distance (HD) model. McEvoy et al. discovered this flaw and propose Isolated WDDL (IWDDL) [11] to solve it. To decrease the fourfold area increase of DWDDL and IWDDL, Baddam and Zwolinski published methods suitable for both ASIC and FPGA. Path Switching [12] balances true and false paths with long high capacitive lines by random swaps. Attractive from a theoretical standpoint, in practice, Path Switching is realized by active elements, which consume power, and thus might diminish robustness against SCA. Divided Backend Duplication [13] consists in splitting dual networks by replacing inverters by exclusive-or (XOR) gates: in precharge phase, the XOR gates are configured as an identity function ($a \mapsto 0 \oplus a = a$) thus propagating the precharge wave, whereas in evaluation phase, they become functional ($a \mapsto 1 \oplus a = \overline{a}$). Unfortunately, this logic has never proved to be glitch-free, nor of constant activity (because of unwanted spurious glitches). Some logic-level upgrades of WDDL have also been proposed, amongst which iMDPL [14], DRSL [6], STTL [15, 16], SecLib [17], WDDL w/o early evaluation [18], and BCDL [19]. All those styles can be mapped onto an FPGA fabric and are thus concerned with dual-rail balancing.

In this article, we do not search to add logic to balance dual networks but rather finely direct the place-and-route (PAR) tool to avoid area increase. The research is led with a view to assess the security level which can be achieved using commercial FPGAs and conclude on their suitability for balanced DPL. Impact of constrained placement of SBoxes has already been described in [20]. Here, we go several steps further: routing is also constrained and we study and experimentally evaluate a complete cryptoprocessor. This latter is presented is Section 2, along with a thorough characterization of the PAR at design-level. Then, we explain how to best take advantage of dual-output programmable logic blocks. The experimental results we obtain against SCA are provided in Section 3. Finally, conclusions and perspectives are discussed in Section 4.

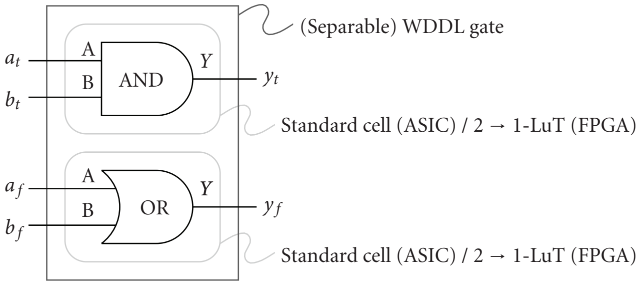

(Separable) WDDL gate
a_t
b_t
A
B
AND
Y
y_t
Standard cell (ASIC) / 2 → 1-LuT (FPGA)
a_f
b_f
A
B
OR
Y
y_f
Standard cell (ASIC) / 2 → 1-LuT (FPGA)

Figure 1: WDDL AND gate, suitable for both ASICs and FPGAs.

2. Fitting WDDL 3DES in Stratix II

2.1. WDDL Roots. In DPL, each bit is represented by a couple of signals, and calculations alternate between precharge phase (PRE) and evaluation phase (EVA). Figure 1 is an example of a WDDL AND gate. In PRE, all of the inputs on the left are forced to “0”, which in turn forces the true output s_t and the false output s_f to “0”. Then, in EVA, after inputs toggle, only s_t or s_f will be set to “1”. This way, whatever the temporal transition, EVA to PRE or PRE to EVA, and whatever the processed data, only one of the differential outputs commutes, yielding a constant transition count. This protocol is illustrated in Figure 2. But this does not suffice to ensure a constant power consumption: the capacitive load of the True and False networks should be the same. Special care should thus be taken for back-end operations, which is the main purpose of this article.

To avoid glitches, a security flaw of WDDL as they are data dependant, synthesis should use only positive functions. For example, logical exclusive OR (XOR) operator cannot be directly used and will be replaced by $a \oplus b = a_t \cdot b_f + a_f \cdot b_t$. This induces cross connections between True and False paths, which increases PAR difficulty.

2.2. WDDL 3DES Cryptoprocessor. The implementation evaluated in this article conforms to the simple and triple Data Encryption Standard (3DES) [21], with all of the specified modes of operations. Although DES has been replaced by the Advanced Encryption Standard (AES) since year 2001, the American National Institute of Standards and Technology (NIST) considers 3DES to be appropriate through year 2030, and the electronic payments industry still uses it for its compactness when implemented in hardware, a quarter of AESs.

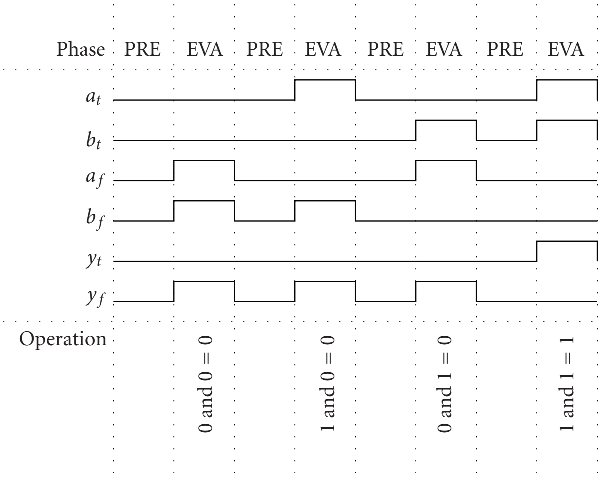


Figure 2: Secure return-to-null protocol for a constant activity of the WDDL AND gate depicted in Figure 1.

The architecture of Figure 3 corresponds to the part of our WDDL 3DES cryptoprocessor which is in charge of processing the 32-bit block R, as defined in page 10 and depicted by page 9 of [21]. This implementation has a straightforward parallel execution, scheduled at one round per clock cycle. The upper and lower paths correspond to the True and False dual networks, respectively. At the beginning of an encryption, the right half of the value resulting of an initial permutation (IP) of the message is loaded into the R master register (R_M), while the R slave register (R_S) still holds the zero precharge value. The latter is applied to the combinatorial logic of the DES datapath, comprised of the following:

(i) the *expansion permutation* (E) function,

(ii) a bit-by-bit addition modulo 2 between the output of E and the round key K_n, referred in the following as XOR_K,

(iii) *Substitution Boxes* (SBoxes) nonlinear functions,

(iv) the *permutation* (P) function,

(v) a bit-by-bit addition modulo 2 between the output of P and the left part of the LR register (L), referred in the following as XOR_L,

which in turn outputs zero. Due to the feedback, at the next rising edge of the clock R_M will load zero while R_S will sample the right half of IP, and the combinatorial logic computes the new intermediate value. This alternates until the end of the encryption.

2.3. Place-and-Route Strategies. Some previous articles [8, 9] already described the mapping of dual-rail logics into FPGA. However, all of them target 4 → 1 look-up-table FPGAs. This choice is interesting from the synthesis point of view, since the mapping is very close to that of an ASIC; as discussed in [4], a simple DPL-compliant design flow is based on the netlist duplication. However, in the FPGA context, keeping two dual instances side-by-side is not trivial; for instance, [22] investigates on the various constraints (for Xilinx & Altera EDA tools) that can be defined at the user-level to force a pairwise placement. A typical mono-output LuT FPGA is the Stratix; Figure 4 illustrates its schematic. Experiments on this target reveal how hard it is to guarantee equal routing for differential signals. For instance, in Figure 5, a complete DES in WDDL is shown. The figure underlines the fact that one net with an extremely high fanout (one bit of the "LR" state addressing the S-Boxes) is doomed to have a different routing, even if the S-Boxes are implemented according to a balanced technique [23]. The idea is that the different location of the "true" and "false" instances will induce a difference in at least one routing path.

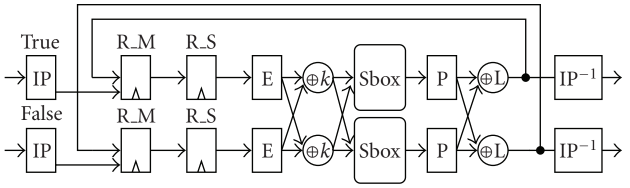


Figure 3: Part of our WDDL 3DES cryptoprocessor processing the 32-bit block R.

Therefore, a native packing of two dual instances into the same reconfigurable resource would be welcome. We choose to assess the robustness of our WDDL 3DES cryptoprocessor when programmed in an Altera Stratix II 90-nm FPGA. Its architecture relies on Adaptive Logic Modules (ALMs), basic building blocks of logic, whose high-level block diagram is shown in Figure 6. On the left, eight inputs drive a combinatorial logic block programmable as either one 6-bit Look-up-Table (6-LuT) or two Adaptive LuTs (ALuTs), both having their own register, named, respectively, *reg0* and *reg1*, on the right.

We have synthesized our implementation using 4-LuT, with the aim to test two PAR strategies. The first one, called "Vertical" strategy, places True and False networks as close as possible by assembling each dual component (R_M, R_S, XOR_K, etc.) in the same ALM (see Figure 7(a)). With the second one, the "Horizontal" strategy, routing resources are identical for the True and False networks: R_M, R_S and XOR_K are packed into the same ALM, and the dual part is placed in the adjacent ALM (see Figure 7(b)).

The floorplan of the Vertical strategy is shown in the layout of Figure 8. The four ALMs on the top left correspond to four bits of R_M. Two dual wires output from each of them, and they all go to R_S, in the middle, then to XOR_K, on the right, and finally reach the SBox and XOR_L (both outside of the figure). The wires on the middle top realize the expansion (E) function towards the SBox number $i+1$, while those on the middle bottom come from the SBox number $i-1$. As explained in Section 2.1, synthesis with positive functions may induce cross connections. Those of XOR_K are clearly visible on the top right of the figure. This schematic is reproduced for each SBox, thus eight times. Post PAR timing annotations give a first idea of the balance between

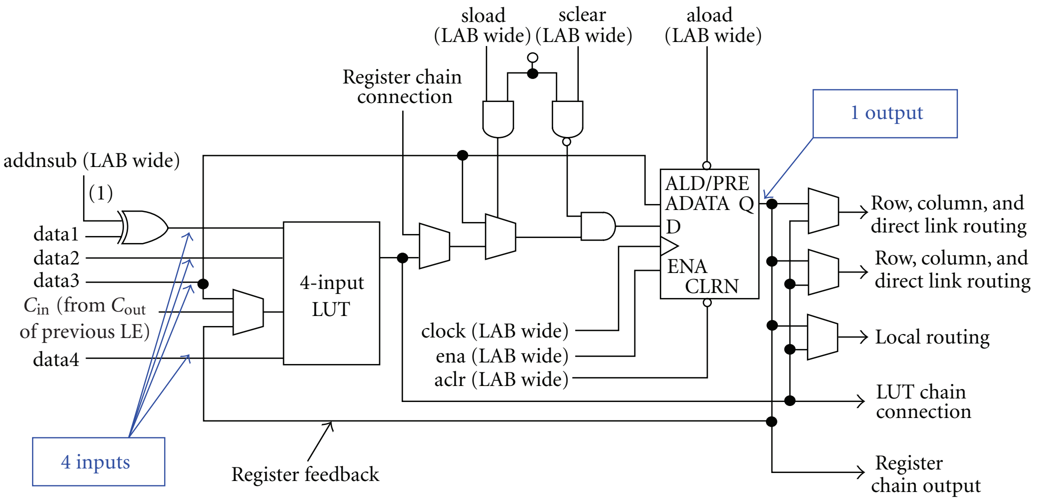


Figure 4: High-Level block diagram of a LuT, extracted from "Stratix Device Family Data Sheet, volume 1" [24].

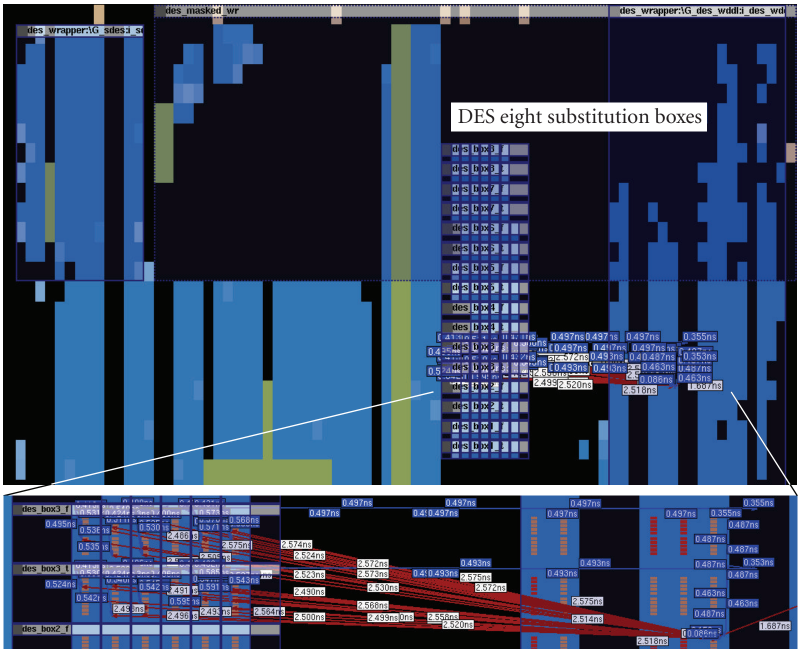

des_masked_wr
des_wrapper:\G_sdes:i_se
des_wrapper:\G_des_wddl:i_des_wd
DES eight substitution boxes
des_box3_f
des_box2_f

Figure 5: Floorplan of a full DES in WDDL (top) and slightly unbalanced fanout of one register of the "LR" state (bottom).

dual wires: for example, the bit of R_M on the top has an imbalance of $289 - 290 = -1$ ps.

Figure 9 illustrates the floorplan for the Horizontal strategy. The upper and lower ALMs contain the true and false networks, respectively. The *reg1* register from Figure 6 is assigned to the master register while the slave register is implemented in *reg0* and XOR_K by the combinatorial logic block, on the left. The gain in balance is optimum as timing annotations have exactly the same value.

2.4. Altera Place-and-Route Constraints. Under Altera's Quartus II design Software, placement is constrained via the LogicLock (LL) regions feature. An LL is a rectangular-shaped region defined by the following:

(1) a *state*: locked at an origin location or floating;

(2) a *size*: fixed user-defined or automatically determined by Quartus;

(3) a *reserved* property: defining whether entities not assigned to this LL region can use its remaining resources.

These properties can be set in a TCL script, according to the syntax shown in Figure 10.

Constraining the routing follows a different procedure. After fitting, a back-annotation of the routing proposed by

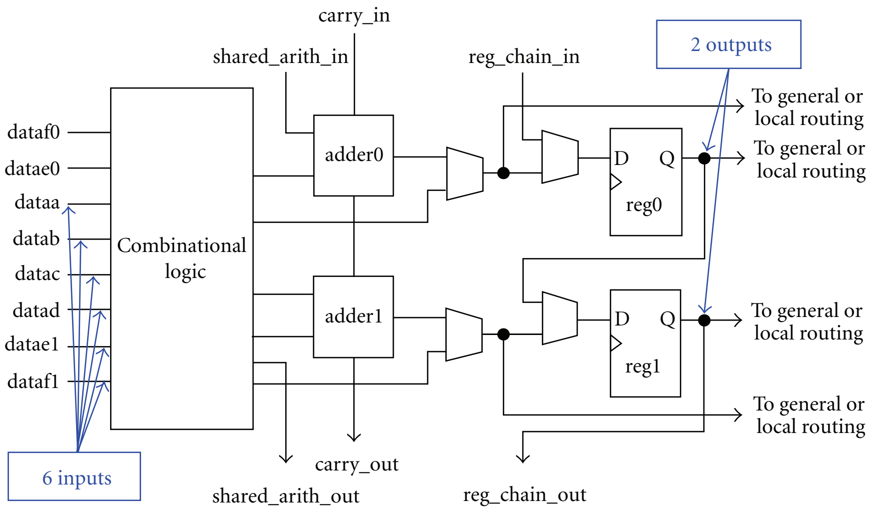
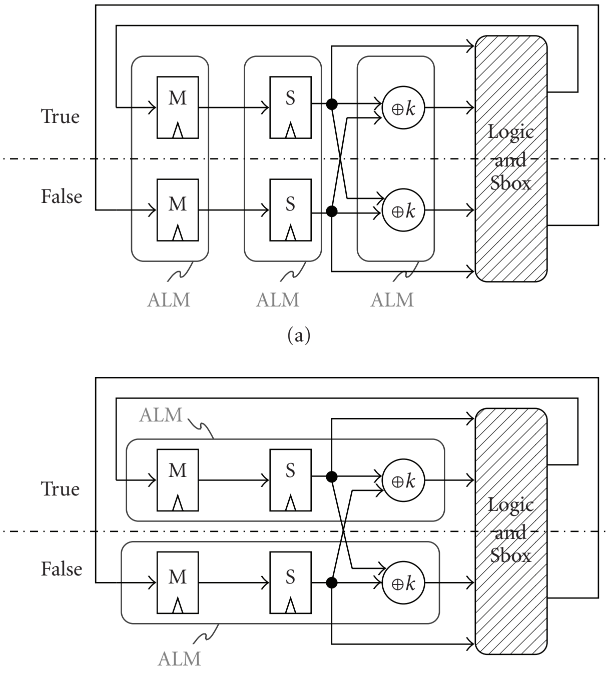


Figure 6: High-Level block diagram of an ALM, extracted from "Stratix II Device Family Data Sheet, volume 1" [24].



(a)

(b)

Figure 7: Vertical (a) and Horizontal (b) PAR strategies.

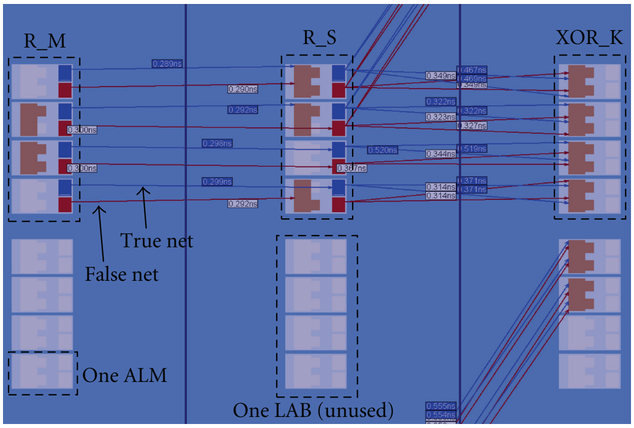


Figure 8: Vertical strategy.

Quartus should be requested. This generates a Routing Constraints File (RCF), whose format is given in Figure 11.

The signal *Input1* of this example comes from the output of the *LE_BUFFER* resource (corresponding to the cluster located at $X = 1$, $Y = 1$, S(ub-location) = 1, I(ndex) = 0) (device-dependant coordinates, described in the "QSF Assignment Descriptions Document"). It then goes through a vertical wire of length 4 (C4) and the local interconnect of the cluster located just above at $X = 1$ and $Y = 2$. The final destination is *DATAC*, an input port on the *InputReg1* block.

To customize the routing, the RCF has to be modified by the user. Afterwards the fitter has to be rerun (with back-annotation enabled to indicate to the Quartus Software that an RCF file exists and should be sourced). Finally, a new back-annotation has to be written out again, in order to verify that the router has met the user-constrained routing.

2.5. Experimental Characterization. Differential traces obtained by DPA can be helpful for a designer to pinpoint countermeasure weaknesses. Indeed, for the right key and monobit analyses, the amplitude of the correlation peak directly gives an experimental evaluation of the power consumption imbalance between true and false networks of a single bit. Many analyses can then be performed. In the following, we present two of them, focusing on the SBox1.

The trace in Figure 11, at the top, corresponds to the power consumption of the FPGA during the beginning of an encryption. At Round 0 evaluation phase, the IP of the message is loaded into LR_S (see Table 1 and Figure 3) after the rising edge of the master clock.

Table 1: Sequence at the beginning of the encryption.

DES Round	Initial State	0 (IP)		1	
WDDL phase	Precharge (PRE)	Precharge (PRE)	Evaluation (EVA)	Precharge (PRE)	Evaluation (EVA)
LR Master	0	IP	0	LR1	0
LR Slave	0	0	IP	0	LR1

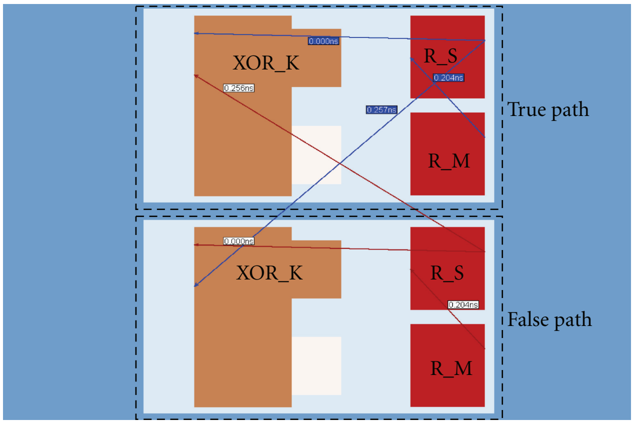


Figure 9: Horizontal strategy.

Table 2: Static evaluation of timing imbalance, in ps.

PAR Strategy	None		Vertical		Horizontal	
Element	Mean	Std Dev	Mean	Std Dev	Mean	Std Dev
LR Master Register	251	322	24	23	3	3
LR Slave Register	566	327	137	88	25	28
XOR_K function	501	298	272	203	290	227
SBox1	202	174	157	119	157	119
SBox2	169	150	169	119	169	119
SBox3	165	155	169	112	169	112
SBox4	146	120	175	123	175	123
SBox5	131	114	167	128	167	128
SBox6	149	154	172	131	172	131
SBox7	170	152	160	118	160	118
SBox8	156	123	173	125	173	125

A few nanoseconds later, the result of the computation is available at the SBoxes output. The differential trace in the middle, obtained by targeting the bit 1 of the SBox1, indeed confirms this. Hence, one correlation peak clearly appears, slightly shifted on the right according to the rising edge of the consumption trace at $t = 418$ ns. Its amplitude, about -35 nV, is negative, suggesting that the false network has a power consumption higher than that of the true network.

Finally, on the rising edge Round 1 precharge phase, the value at the output of XOR_K is sampled by LR_M. Once again, a correlation peak emerges in the differential trace in the bottom, generated when targeting bit 16 of LR_M (after the P permutation, the bit 1 of SBox1 becomes bit 16 of LR). This time, the amplitude of the peaks is close to -60 nV. As DPA looks for the maximum amplitude, attacks on the LR register are the most powerful in our case.

This phenomenon can either be observed with single rail logic: indeed, registers consume more power than combinatorial logic.

Here, the explanation is that in combinatorial logic, commutation dates are data-dependant. Thus, the power consumption of each encryption vanishes in the differential trace because of a noncoherent averaging due to the intrinsic DPA processing whereas the power consumption of the register is well synchronized.

To gather it, we have to try to use a 4th-order integration [25], unsuccessfully.

To close these analyses, we have reported in Figure 10 the amplitude of the SBox1 output bits correlation peaks when targeting LR. Bit 2 has the highest amplitude, which justifies that it delivers the best performance for the attack (see Table 3). The imbalance is positive for the first and last bits, negative for the others. This experimentally confirms that attacking four bits at once as with unprotected module is less efficient, because bit imbalances counterbalance each other. An improved attack against DPL may be summing up the absolute value of each correlation peak.

2.6. Static Evaluation of the Place-and-Route. In terms of timing differences, the study of the differential PAR quality can be achieved with an analysis of the Standard Delay Format (SDF) file generated by the Quartus II Compiler. Such a file gives interconnection and propagation delays of each instance in the FPGA. Analysis of registers and XOR function timing delays is trivial as less than three couples of dual wires per bit have to be considered. Determining the imbalance in the SBoxes is more difficult; indeed, all delay differences along the datapath have to be summed up.

Mean and standard deviation statistics on timing differences between the true and false network are given in Table 2 in picoseconds. The first column corresponds to the imbalance without specific PAR constraints. It serves as a reference to show the improvement generated by the two PAR strategies. On average, timing imbalances of SBoxes are not much affected by PAR strategies while those of the XOR_K are halved. The Horizontal strategy has a great impact on register imbalance, diminishing them to 3 ps. If the robustness is strongly correlated to the timing differences, this strategy should be the most robust. The next section on experimental results of the robustness evaluation shows that the opposite holds true.

```
// Definition of a LogicLock region assigned to <entity>,
// with STATE = locked at (1;1), SIZE = 10 LAB × 20 LAB,
// and free resources available for others entities.
set_global_assignment -name LL_MEMBER_OF <entity>
set_global_assignment -name LL_STATE LOCKED
set_global_assignment -name LL_ORIGIN LAB_X1_Y1
set_global_assignment -name LL_AUTO_SIZE OFF
set_global_assignment -name LL_WIDTH 10
set_global_assignment -name LL_HEIGHT 20
set_global_assignment -name LL_RESERVED OFF
```

Figure 10: Example of TCL script setting up a LogicLock region.

```
signal_name = Input1  {
LE_BUFFER:X1Y1S1I0;
C4:X1Y1S0I25;
LOCAL_INTERCONNECT:X1Y2S0I15;
dest = ( InputReg1, DATAC );
}
```

Figure 11: Example of an RCF file reporting the routing resources usage.

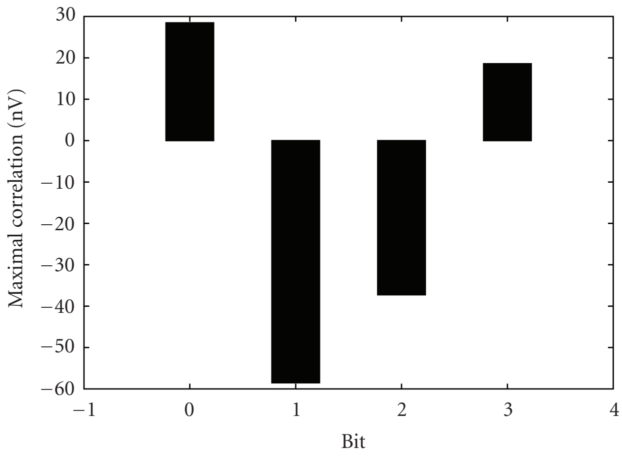


Figure 12: Histogram of SBox1 output bits power imbalance.

3. Security Evaluation

3.1. Background Material. Security evaluation is extremely influenced by various parameters: setups, acquisition conditions, target algorithms, target board, attack models, and so forth. Thus, it is hard to compare SCA countermeasures and attacks of different laboratories on a fair ground. To contribute to the elaboration of common platforms, guaranteeing experiments reproducibility for research institutions, the Tohoku University, and the Japanese Research Center for Information Security (RCIS) has developed in 2007 "Side-channel Attack Standard Evaluation Boards" (SASEBOs) [26]. Dedicated to the evaluation of SCA countermeasures, these electronic boards are distributed free of charge to academic laboratories leading innovative researches in the field of embedded security. Thereby, everyone can reproduce, analyse, and criticize results of their peers.

There are four versions, depending on the target chip. All experiments in this paper have been realized using the SASEBO-B, which incorporates two Altera [24] Stratix II FPGA: one *EP2S30F672C5N* supposed to embed all control modules, and one *EP2S15F484C5N* for the cryptographic modules. Having two FPGAs enhances the accuracy of the measurements as it uncouples the power consumption of the cryptographic parts from the power consumption of the others. Another measurement improvement feature is the possibility of using separate power supplies for the core and the input and output pads of the FPGA. Measurements have been done using the original 1 Ω spying shunt resistor in the positive rail of the cryptographic FPGA core power supply. The acquisition board is depicted in Figure 12.

To improve experiment reproducibility by peers, we have used "Eve (Eavesdropper) SoC", a System on Chip (SoC) providing a flexible way to easily and rapidly design real-life cryptographic applications: when a stand-alone hardware module protected by an SCA countermeasure is fully functional, it can simply be bound to EveSoc as a plug-and-play custom coprocessor, so as to constitute a complete cryptographic device. VHDL code files of EveSoc along with its documentation and miscellaneous tools including scripts for SASEBOs are freely downloadable from its *SourceForge* development website [27].

Power consumption measurements, referred in the following as *traces*, have been collected using a *1132A* differential probe and a *54855 Infiniium* oscilloscope from Agilent Technologies. The final setup has a 6 GHz bandwidth and a 40 GSa/s maximal sample rate.

3.2. Security Gain. As explained in the above section, various parameters affect the security evaluation, and it is always difficult to determine whether the hardness of an attack is caused by a good countermeasure or by a weak adversary. In this context, a framework has been proposed in [28] which suggests to start with an information theoretic analysis to assess the maximal amount of information which could be extracted. This corresponds in practice to the worst case attack, which is difficult to devise in the case of hardware implementations. Then, various distinguishers have to be used to see how a given adversary can take advantage of the

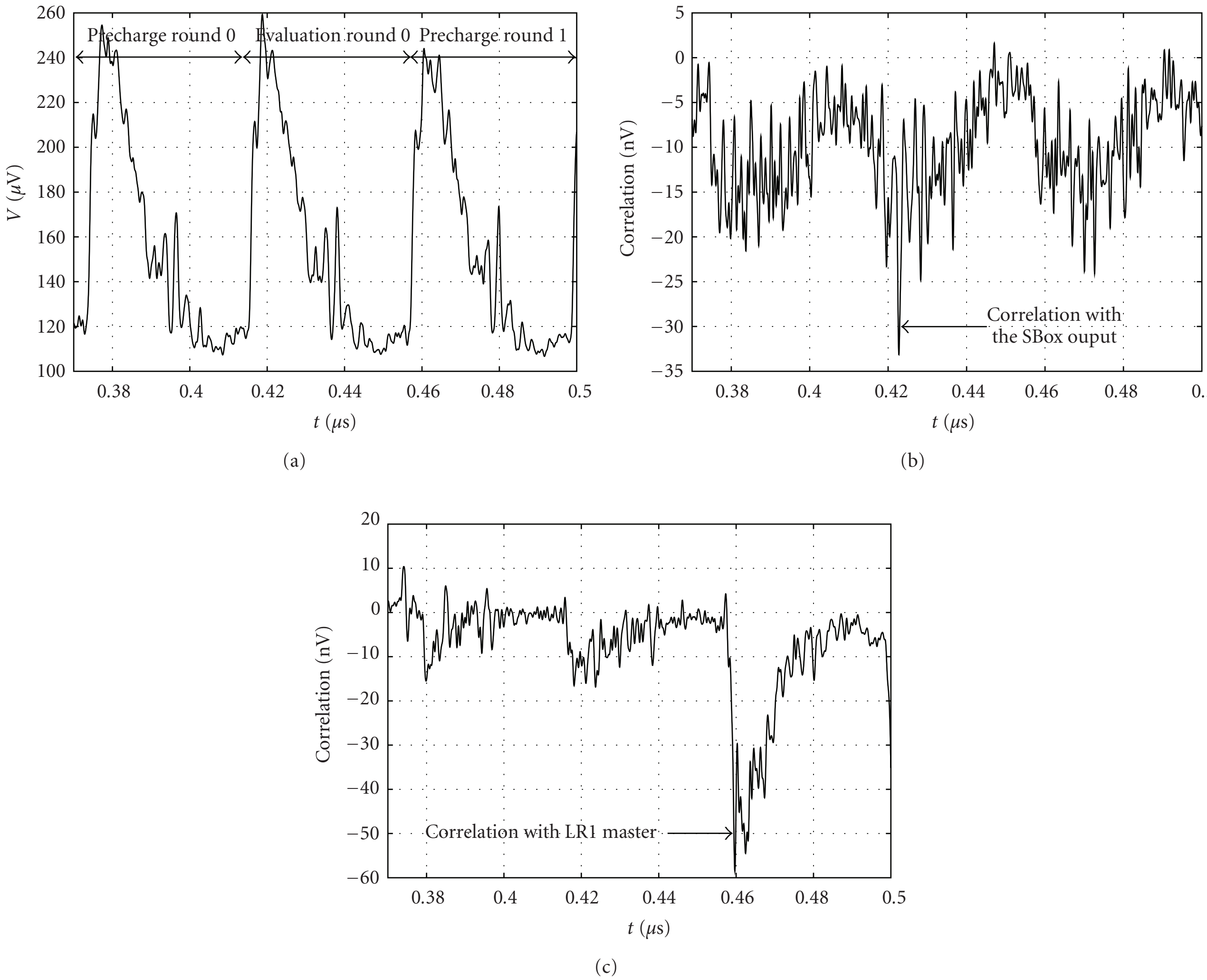


Figure 13: From (a)–(c) single trace and differential traces when targeting the SBox1 and the LR registers.

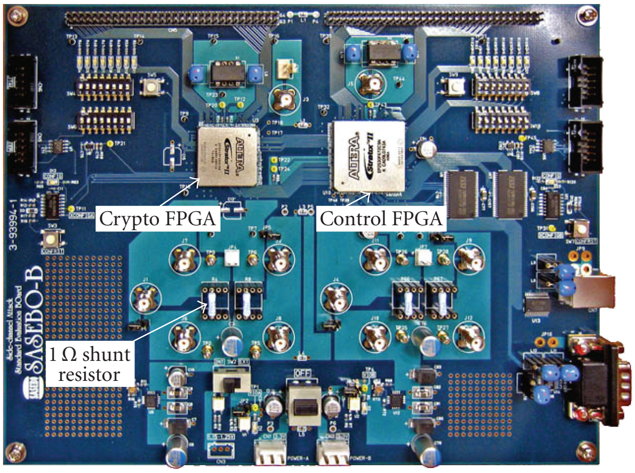


Figure 14: SASEBO-B configuration for the DPA attacks.

Table 3: Statistics for Bits of R_S Register.

(a) WDDL 3DES Module without PAR Constraints

SBox	1				2				3				4			
Bit of R	9	17	23	31	13	28	2	18	24	16	30	6	26	20	10	1
Timing imbalance, in ps	266	917	796	308	331	885	265	633	560	1,073	286	840	599	611	851	**253**
Security Gain	3	1	5	1	1	6	3	1	2	11	10	6	2	4	2	1
Minimal Security Gain	1				1				2				1			
SBox	5				6				7				8			
Bit of R	8	14	25	3	4	29	11	19	32	12	22	7	5	27	15	21
Timing imbalance, in ps	290	686	944	1,014	1,574	261	700	666	306	340	865	262	322	1,032	348	621
Security Gain	1	2	5	7	4	2	8	1	3	**22**	2	4	1	1	3	2
Minimal Security Gain	1				1				2				1			

(b) WDDL 3DES Module with Vertical Strategy

SBox	1				2				3				4			
Bit of R	9	17	23	31	13	28	2	18	24	16	30	6	26	20	10	1
Timing imbalance, in ps	38	64	23	23	53	30	5	1	29	32	**0**	1	1	59	1	17
Security Gain	50	10	9	5	1	8	10	4	12	20	9	31	6	21	23	21
Minimal Security Gain	5				1				9				6			
SBox	5				6				7				8			
Bit of R	8	14	25	3	4	29	11	19	32	12	22	7	5	27	15	21
Timing imbalance, in ps	25	1	29	1	123	30	1	1	51	9	1	**0**	42	1	59	42
Security Gain	2	19	13	—	214	5	15	—	19	352	11	31	36	1	8	31
Minimal Security Gain	2				5				11				1			

(c) WDDL 3DES Module with Horizontal Strategy

SBox	1				2				3				4			
Bit of R	9	17	23	31	13	28	2	18	24	16	30	6	26	20	10	1
Timing imbalance, in ps	325	125	125	152	182	176	**14**	29	156	146	65	41	38	221	42	355
Security Gain	9	8	7	2	3	10	4	1	18	15	25	11	2	5	20	5
Minimal Security Gain	2				1				15				2			
SBox	5				6				7				8			
Bit of R	8	14	25	3	4	29	11	19	32	12	22	7	5	27	15	21
Timing imbalance, in ps	26	39	204	136	168	139	147	247	208	166	14	150	261	50	83	182
Security Gain	2	4	2	15	110	8	11	15	23	**364**	18	23	32	4	9	14
Minimal Security Gain	2				8				18				4			

information leakage. However, with hardware implementation such as the one studied in this article, the construction of templates and the information theoretic analysis is very intensive and may not lead to better results than directly performing an attack. Thus, for a first study, we prefer to limit the security analysis and focus on few meaningful attacks.

The strength of these attacks, and thus the robustness against SCA of an implementation which is attacked, was first quantified by evaluating how many Measurements to Disclose (MTD) the secret key are needed. But this number has appeared to be dependent on the values, on the order of the messages, and on the secret key. Therefore, a sound approach is to perform a lot of attacks with different keys, for instance, one hundred, and then to consider the MTD for a given Global Success Rate (GSR), saying 90% or 95%. However, this absolute MTD remains not significant for a firm conclusion. Indeed, what could we think of a countermeasure which needs 1,500,000 traces to be broken, whereas its unprotected version resists analyses up to 1,000,000 measurements? Thus, in the following, we will quantify the security robustness by computing the *security gain* (SG), that is, the ratio between the MTD of a protected module and the MTD of an unprotected module:

$$\mathrm{SG} = \frac{\text{MTD-GSR90\%}_{\text{protected}}}{\text{MTD-GSR90\%}_{\text{unprotected}}}. \qquad (1)$$

3.3. Experimental Results. With a view to cover a large SCA threat range, we have acquired 6,400,000 traces and performed numerous analyses: Differential Power Analysis (DPA) [29], but also Correlation Power Analysis (CPA) [30], with both Hamming Weight (HW) and Hamming Distance (HD) models, by guessing one to four bits of the main internal states of the algorithm, that is, the output values of R (register), XOR_K, SBoxes, but not XOR_L as it is tantamount to guessing R. We discard other analyses, such as the template attacks [31], as they have never proved to be practical on parallel implementations of block ciphers. Besides, Mutual Information Analysis (MIA) [32] outperforms CPA [33] only when the side-channel signals are noisy, and leakages multiple, which is not our case. Moreover, we restrict the number of hypotheses to those on the first round key, seen as eight classes of 6-bit each. Hence, an attack on the unprotected module using the HD can only concern R.

Once again, experiments confirm that CPA is more powerful than the DPA in presence of noise (traces have not been averaged) [34]. Best results on unprotected implementations are obtained with the HD model, as it matches the physical phenomenon responsible for power consumption (commutations), and by targeting four bits at the same time, increasing the signal-to-noise (SNR) ratio of the correct peak with respect to the peaks present in incorrect guesses. For WDDL, because of the zero value precharge, best analyses are done by guessing the HW as it equals the number of commutations: $\mathrm{HD}(0, x) = \mathrm{HW}(x)$. Targeting a single bit is more powerful than four bits: indeed, the leakage in WDDL is caused by the imbalance between the True and the False networks, and this unbalance could be the opposite for targeted bits and therefore counterbalance each other. The same observation has been basically done for quasidelay insensitive asynchronous circuits [35].

We observe that no PAR strategy resists SCA attacks. As the weak element of the implementation is R_S, we focus in the following on the robustness of each bit of this register. Table 3 summarizes the results for all of the three modules, which are truly comparable as traces have been acquired using the same experimental setup, and the same pseudorandom messages, generated from the same seed. Table 3(a) concerns an unconstrained WDDL cryptoprocessor, serving as reference to estimate the security gain provided by WDDL, and to study the impact of the differential PAR. Tables 3(b) and 3(c) deal, respectively, with the results of Vertical and Horizontal PAR strategies. For Tables 3(a), 3(b), and 3(c), bits coming from the same SBox have been grouped together, and their position in the R register after (P) permutation of 3DES is recalled by the second and sixth lines. For example, bit 1 of SBox 1 becomes bit 9 of R. The third and seventh lines correspond to the timing imbalance in picoseconds. They detail values of Table 2. “Zero” means under the resolution of the timing analysis tool (one picosecond). Fourth and eighth lines provide the SG defined in Section 3.2. The HW model monobit attack has been one hundred times reiterated to get a representative success rate. The SG is then computed with the minimal number of traces needed to recover the full secret key (100% success rate, worst case).

Boldface cells correspond to best cases, that is, smallest value for the timing imbalance and largest value for the SG. The presence of two boldface cells in a same column means that static and experimental evaluations are correlated. An overall look at Table 3 shows that it is never the case. This tends to confirm that evaluating the imbalance with delays is not accurate enough, as they are maximum values, and not typical. One solution to improve analysis may be to rather consider the type of the line, short or long, and to take into account the fanout.

Best cases show that SG is increased by at most a factor 22 when using WDDL without specific efforts of PAR, 364 when using the Horizontal PAR strategy. The Vertical PAR strategy seems to be the more robust, as bit 31 and bit 18, marked with black shaded cells, do not disclose the secret key using 6,400,000 messages. Unfortunately, a reliable security assessment considers the worst case. With 3DES, SG of one SBox equals the minimal SG amongst the SG of its four output bits. Thus, a more accurate conclusion is that the WDDL implementation without specific efforts of PAR increases the robustness a little; the Horizontal PAR strategy multiplies it by 18 to break the SBox 7, but by 6.5 on average; the SG of the Vertical PAR strategy equals 11 for the SBox 7, 5 on average. Referring to Table 1 of [4], MTDs of the unprotected and WDDL-protected modules of Kris Tiri equal 320 and 21,185, respectively, discarding the 5 key bytes not disclosed. The SG is thus close to 66, overriding by one order of magnitude that of our FPGA implementation. However, some bits of the Vertical Strategy present a high robustness: two of them do not result in a disclosure, and two others have an SG of 214 and 352. These results are very promising: we think that we could in the future increase the SG of all of the bits to this level. A reliable approach to reach this goal seems to be in-depth analyses of the link between the routing resources used by a differential path and the SG of its corresponding bit.

4. Conclusion and Perspectives

We have presented in this article a WDDL 3DES implementation with a fully-fledged placement and routing reaching a timing balance lower than 290 ps according to static evaluation. The unconstrained DPL design is shown to be definitely more secure than the reference design. In addition, the PAR strategies we introduced increase further the secure level on top of the unconstrained design. However, despite these encouraging results, the implementations are still attackable with SCA, albeit with more measurements, thus making the attack more difficult.

A detailed analysis has shown that weaknesses originate from register imbalance, although, up to now, efforts were devoted to the design of SBoxes. Security evaluation cannot rely on experiments on a single cryptographic primitive and should concern a complete and real-life application to avoid side effects, by means of solutions such as EveSoc and SASEBO.

Efficient differential place-and-route on FPGA appears to be a great challenge. Perspectives for future works are to validate the influence of technological characteristic variability over the same chip by renewing experiments at different locations, but on another FPGA families as well. We plan also to test other FPGA design solutions, such as Xilinx’s ISE, which may propose more accurate timing analysis tools.

References

[1] L. Goubin and J. Patarin, "DES and differential power analysis," in *Workshop on Cryptographic Hardware and Embedded Systems*, LNCS, pp. 158–172, Springer, worcester, Mass,USA, Aug 1999.

[2] J. Blömer, J. Guajardo, and V. Krummel, "Provably secure masking of AES," in *Proceedings of the Selected Areas in Cryptography (SAC '04)*, vol. 3357 of *LNCS*, pp. 69–83, Springer, Waterloo, Canada, August 2004.

[3] É Peeters, F.-X. Standaert, N. Donckers, and J.-J. Quisquater, "Improved higher-order side-channel attacks with FPGA experiments," in *7th International Workshop on Cryptographic Hardware and Embedded Systems, CHES 2005*, pp. 309–323, gbr, September 2005.

[4] K. Tiri and I. Verbauwhede, "A digital design flow for secure integrated circuits," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 25, no. 7, pp. 1197–1208, 2006.

[5] T. Popp and S. Mangard, "Masked dual-rail pre-charge logic: DPA-resistance without routing constraints," in *Workshop on Cryptographic Hardware and Embedded Systems*, vol. 3659 of *LNCS*, pp. 172–186, Springer, Edinburgh, UK, August 2005.

[6] C. Zhimin and Z. Yujie, "Dual-rail random switching logic: a countermeasure to reduce side channel leakage," in *Workshop on Cryptographic Hardware and Embedded Systems*, vol. 4249 of *LNCS*, pp. 242–254, Springer, Yokohama, Japan, 2006.

[7] P. Schaumont and K. Tiri, "Masking and dual-rail logic don't add up," in *Workshop on Cryptographic Hardware and Embedded Systems*, vol. 4727 of *LNCS*, pp. 95–106, Springer, Vienna, Austria, September 2007.

[8] K. Tiri and I. Verbauwhede, "Synthesis of secure FPGA implementations," in *Proceedings of the International Workshop on Logic and Synthesis (IWLS '04)*, pp. 224–231, June 2004.

[9] 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 *Proceedinsg of the Conference on Secure Software Integration and Reliability Improvement (SSIRI '08)*, pp. 16–23, IEEE Computer Society, Yokohama, Japan, July 2008.

[10] P. Yu and P. Schaumont, "Secure FPGA circuits using controlled placement and routing," in *Proceedings of the 5th IEEE/ACM International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS '07)*, pp. 45–50, ACM, New York, NY, USA, 2007.

[11] R. P. McEvoy, C. C. Murphy, W. P. Marnane, and M. Tunstall, "Isolated WDDL: a hiding countermeasure for differential power analysis on FPGAs," *ACM Transactions on Reconfigurable Technology and Systems*, vol. 2, no. 1, pp. 1–23, 2009.

[12] K. Baddam and M. Zwolinski, "Path switching: a technique to tolerate dual rail routing imbalances," *Design Automation for Embedded Systems*, vol. 12, no. 3, pp. 207–220, 2008.

[13] K. Baddam and M. Zwolinski, "Divided backend duplication methodology for balanced dual rail routing," in *Workshop on Cryptographic Hardware and Embedded Systems*, vol. 5154 of *LNCS*, pp. 396–410, Springer, Washington, DC, USA, August 2008.

[14] T. Popp, M. Kirschbaum, T. Zefferer, and S. Mangard, "Evaluation of the masked logic style MDPL on a prototype chip," in *Workshop on Cryptographic Hardware and Embedded Systems*, vol. 4727 of *LNCS*, pp. 81–94, Springer, Vienna, Austria, September 2007.

[15] R. Soares, N. Calazans, V. Lomné, P. Maurine, L. Torres, and M. Robert, "Evaluating the robustness of secure triple track logic through prototyping," in *Proceedings of the 21st Annual Symposium on Integrated Circuits and Systems Design (SBCCI '08)*, pp. 193–198, ACM, New York, NY, USA, September 2008.

[16] V. Lomné, P. Maurine, L. Torres, M. Robert, R. Soares, and N. Calazans, "Evaluation on FPGA of triple rail logic robustness against DPA and DEMA," in *Proceedings of the Design, Automation and Test in Europe (DATE '09)*, pp. 634–639, IEEE, Nice, France, April 2009, track A4 (Secure embeddedimplementations).

[17] S. Guilley, F. Flament, R. Pacalet, P. Hoogvorst, and Y. Mathieu, "Security evaluation of a balanced quasi-delay insensitive library," in *Proceedings of the Design of Circuits and Integrated Systems (DCIS '08)*, p. 6, IEEE, Grenoble, France, November 2008, Session 5D—Reliable and Secure Architectures.

[18] S. Bhasin, J.-L. Danger, F. Flament et al., "Combined SCA and DFA countermeasures integrable in a FPGA design flow," in *Proceedings of the International Conference on ReConFigurable Computing and FPGAs (ReConFig '09)*, pp. 213–218, IEEE Computer Society, Quintana Roo, México, December 2009.

[19] M. Nassar, S. Bhasin, J.-L. Danger, G. Duc, and S. Guilley, "BCDL: a high speed balanced DPL for FPGA with global precharge and no early evaluation," in *Proceedings of the Design, Automation and Test in Europe (DATE '10)*, pp. 849–854, IEEE Computer Society, Dresden, Germany, March 2010.

[20] S. Guilley, S. Chaudhuri, L. Sauvage et al., "Place-and-route impact on the security of DPL designs in FPGAs," in *Proceedings of the IEEE International Workshop on Hardware-Oriented Security and Trust (HOST '08)*, pp. 26–32, IEEE, Anaheim, Calif, USA, June 2008.

[21] DES, 1999, http://csrc.nist.gov/publications/fips/fips46-3/fips46-3.pdf.

[22] S. Guilley, S. Chaudhuri, L. Sauvage et al., "Place-and-route impact on the securityof DPL designs in FPGAs," in *Proceedings of the IEEE International Workshop on Hardware-Oriented Security and Trust (HOST '08)*, pp. 26–32, IEEE, Anaheim, Calif, USA, June 2008.

[23] T. Akishita, M. Katagi, Y. Miyato, A. Mizuno, and K. Shibutani, "A practical DPA countermeasure with BDD architecture," in *Proceedings of the International Conference on Smart Card Research and Advanced Application (CARDIS '08)*, vol. 5189 of *Lecture Notes in Computer Science*, pp. 206–217, Springer, London, UK, September 2008.

[24] "Altera FPGA designer," http://www.altera.com.

[25] T.-H. Le, J. Clédière, C. Servière, and J.-L. Lacoume, "Noise reduction in side channel attack using fourth-order cumulant," *IEEE Transactions on Information Forensics and Security*, vol. 2, no. 4, pp. 710–720, 2007.

[26] SASEBO, http://www.rcis.aist.go.jp/special/SASEBO/index-en.html.

[27] SourceForge, https://sourceforge.net/projects/evesoc/.

[28] F.-X. Standaert, T. G. Malkin, and M. Yung, "A unified framework for the analysis of side-channel key recovery attacks," in *Proceedings of the Annual International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT '09)*, vol. 5479 of *LNCS*, pp. 443–461, Springer, Cologne,Germany, April 2009.

[29] P. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," in *Proceedings of the International Cryptology Conference (CRYPTO '99)*, vol. 1666 of *LNCS*, pp. 388–397, Springer, Santa Barbara, Calif, USA, Augudt 1999.

[30] É Brier, C. Clavier, and F. Olivier, "Correlation power analysis with a leakage model," in *Workshop on Cryptographic Hardware and Embedded Systems*, vol. 3156 of *LNCS*, pp. 16–29, Springer, Cambridge, Mass, USA, August 2004.

[31] S. Chari, J. R. Rao, and P. Rohatgi, "Template attacks," in *Workshop on Cryptographic Hardware and Embedded Systems*, vol. 2523, pp. 13–28, Springer, Redwood City, Calif, USA, August 2002.

[32] B. Gierlichs, L. Batina, and P. Tuyls, "Mutual information analysis—a universal differential side-channel attack," report 2007/198, 2007, http://eprint.iacr.org.

[33] E. Prouff and M. Rivain, "Theoretical and practical aspects of mutual information based side channel analysis," in *Proceedings of the conference on Applied Cryptography and Network Security (ACNS '09)*, vol. 5536 of *LNCS*, pp. 499–518, Springer, Paris-Rocquencourt, France, June 2009.

[34] S. Guilley, L. Sauvage, P. Hoogvorst, R. Pacalet, G. M. Bertoni, and S. Chaudhuri, "Security evaluation of WDDL and SecLib countermeasures against power attacks," *IEEE Transactions on Computers*, vol. 57, no. 11, pp. 1482–1497, 2008.

[35] G. Bouesse, M. Renaudin, B. Robisson et al., "DPA on quasi delay insensitive asynchronous circuits: concrete results," in *Proceedings of the 19th Conference on Design of Circuits and Integrated Systems (DCIS '04)*, pp. 24–26, Bordeaux, France, November 2004.

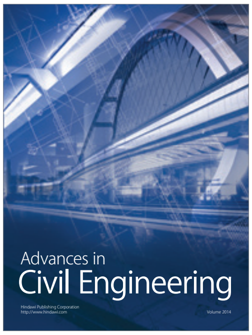
Journal of
Engineering

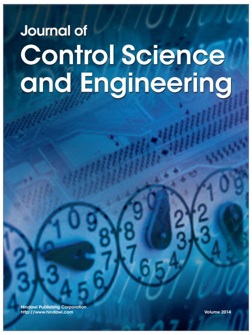
The Scientific
World Journal

International Journal of
Rotating
Machinery

Journal of
Sensors

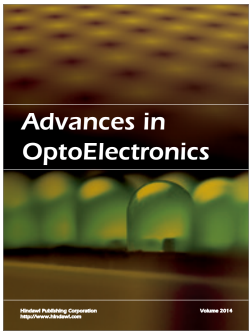
International Journal of
Distributed
Sensor Networks


Advances in
Civil Engineering


Journal of
Control Science
and Engineering

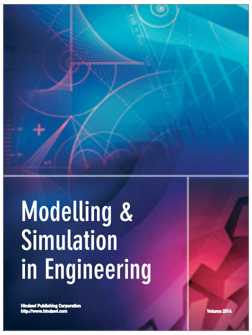
Journal of
Robotics

Journal of
Electrical and Computer
Engineering

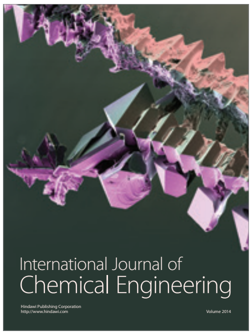

Advances in
OptoElectronics

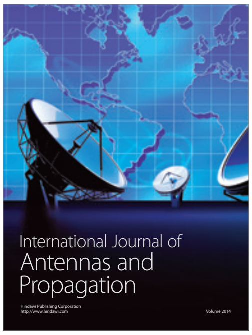
VLSI Design

International Journal of
Navigation and
Observation

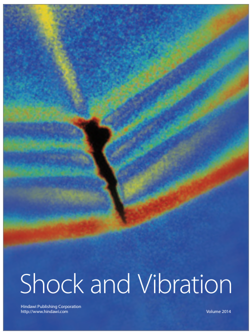

Modelling &
Simulation
in Engineering

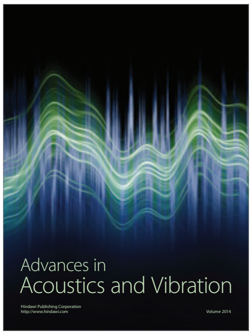
International Journal of
Aerospace
Engineering


International Journal of
Chemical Engineering


International Journal of
Antennas and
Propagation

Active and Passive
Electronic Components


Shock and Vibration


Advances in
Acoustics and Vibration