Experimental Evaluation of Protections Against Laser-induced Faults and Consequences on Fault Modeling

R. Leveugle, A. Ammari, V. Maingot
TIMA Laboratory - 46 Avenue Félix Viallet - 38031 Grenoble Cedex - France
E. Teyssou
Thalès Communications - 160 Boulevard de Valmy, BP 82 - 92704 Colombes Cedex - France
P. Moitrel, C. Mourtel, N. Feyt
Gemalto - La Vigie, Avenue du Jujubier, ZI athelia IV - 13705 La Ciotat Cedex - France
J.-B. Rigaud*, A. Tria†
*EMSE †CEA-LETI - SESAM Laboratory - CMPGC, rue des Anémones - 13541 Gardanne - France

Abstract

Lasers can be used by hackers to situations to inject faults in circuits and induce security flaws. On-line detection mechanisms are classically proposed to counter such attacks, and are often based on error detecting codes. However, the efficiency of such schemes has not been precisely validated against real attack conditions. This paper presents results showing that, with a given type of laser, a classical protection technique can leave open doors to an attacker. The results give also insights into the fault models to be taken into account when designing a secured circuit.

1. Introduction

Lasers are used in several situations to inject faults in circuits. One of the applications is to evaluate the reaction of a circuit to a very local perturbation, intended to be representative of a particle impact [1]. Publications on laser-induced faults are almost limited to this type of characterization of circuits used in harsh environments such as space. However, lasers can also be used by hackers to induce security flaws. In this context, the goal is to perturb the normal behavior so that unauthorized privileges are granted or secret information can be retrieved. Such fault-based attacks have become one of the main threats for circuits designed with security constraints [2, 3], motivating a lot of work on protection schemes (e.g. [4, 5]). Other techniques can be used to inject faults in a circuit, including glitches on the clock or power lines and perturbations by white light (using e.g. a photoflash lamp). However, the laser is much more powerful due to the possibility to directly perturb an internal area of the circuit with accurate fault injection area and time specification.

In this paper, we are mainly interested in studying the effects of faults induced to perform attacks. The very sophisticated lasers used to create effects similar to particle impacts are therefore not our main concern. As a matter of fact, such lasers are very costly, and therefore quite difficult to acquire. Also, the energy levels are quite low, often requiring using complex backside techniques to inject faults in a circuit in spite of the multiple metallic interconnection levels used in up-to-date technologies. A hacker may therefore prefer to use less costly equipments, with limited focus capabilities, but allowing him to perturb the circuit behavior with a simple front-side shot, only requiring opening the circuit package.

On-line detection mechanisms are classically proposed to counter such attacks, and are often based on detecting codes to achieve low overheads [5]. However, the efficiency of such an approach has not been precisely validated against real attack conditions. More precisely, experiments are carried out by the manufacturers to qualify their products, but the results remain an industrial secret. Only a few studies have been very recently published, giving some insights into the effects of laser shots on asynchronous logic [6] or on specific logic patterns designed to analyze gate sensitivity [7]. The aim of this work is to analyze the effects of a laser shot on a sequential synchronous circuit with significant complexity, to evaluate the efficiency of a classical error detection scheme, and to derive information about the faults induced by a standard laser source. Accurate fault models are indeed necessary to design efficient protections. For this purpose, a circuit has been designed

978-3-9810801-2-4/DATE07 © 2007 EDAA

and manufactured with two versions: a reference version without protections and a protected version. A laser fault injection campaign has then been carried out in the Gemalto facilities and the main results are summarized.

The reference circuit characteristics are summarized in section 2. The protections implemented in the second version are presented in section 3. The experimental set-up is described in section 4. Finally, results are discussed in section 5.

2. Case study: a Montgomery coprocessor

2.1. Functional specification

The circuit chosen as test vehicle is a coprocessor core performing Montgomery multiplications. This core can be used as a hardware accelerator to crypt and decrypt messages using the RSA algorithm.

The Montgomery multiplier computes $A*B*R^{-1}$ Mod N with A, B and N coded on the same number n of bits and $R=2^n$. A and B must be smaller than N and N must be odd. In the case of the implemented circuit, n=512.

2.2. Global architecture

The multiplier is described at Register Transfer level in VHDL. The main blocks in the description are summarized in Figure 1. The computation core is a systolic combinatorial array of 512 cells, each of them built with adders and a few logic gates. This array is controlled by a sequential logic and fed by a shift register allowing to shift the value of the A operand each two cycles. The input data (A, B, N) and the result are stored in 512-bit I/O registers, connected to the system through a simple AHB-Lite interface. The Montgomery multiplier can then be used as a slave peripheral when connected to an AMBA bus [8]. Data are transferred using 32-bit sub-words, assembled in the I/O registers.

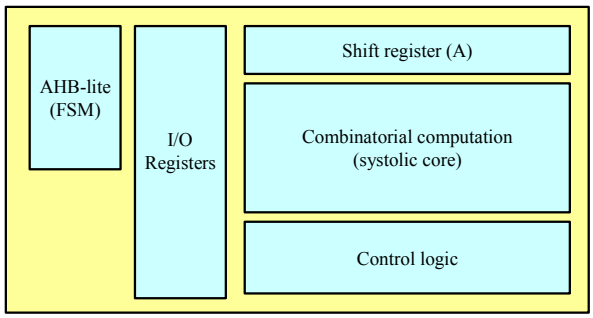

Figure 1: Main blocks in the RTL description.

The main operation phases are summarized in Figure 2. 3*n=1536 clock cycles are necessary to complete the computation in the systolic array, after data loading.

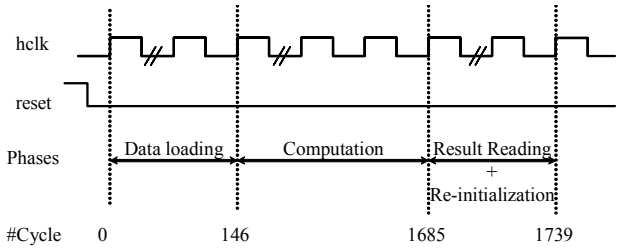

Figure 2: Operation phases.

3. Implemented protections

3.1. Protection techniques

All registers in the circuit have been protected using parity encoding and dual-rail checkers. In order to achieve a good trade-off between overheads and multiple fault detection, a parity bit has been added to each 32-bit sub-word. In consequence, 16 parity bits are added to a register storing 512-bit data. Small registers or individual flip-flops have been protected by one parity bit for each group of 32 flip-flops. The state register of the AHB interface is a particular case, protected using a one-hot state assignment.

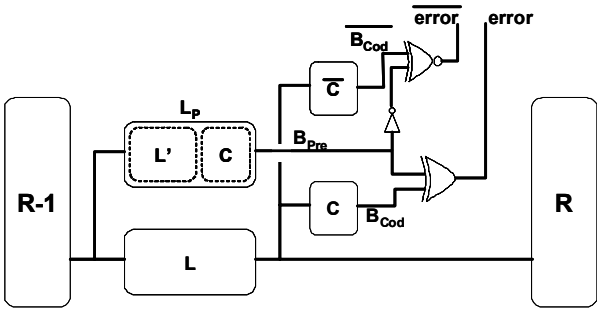

Figure 3: Basic approach for RT-level parity prediction implementation.

The combinatorial computation array has been protected using a parity prediction scheme applied at RT-Level. The general approach is presented in [9] and is illustrated in Figure 3 where R-1 and R are respectively the input and output register of a combinatorial logic L. A prediction block Lp is added, just made of the assembling of a replica of L, called L', and the coder C. The outputs of this block are the predicted code bits BPre. The prediction block is described at the same level of hierarchy as the block L, and the two blocks are synthesized independently. In that way, the block Lp can be in most cases noticeably simplified since the outputs of L' are only intermediate signals. On the opposite, the outputs of L are connected to R and cannot be simplified. The correctness of the outputs of L is then checked by computing their check bits BCod and comparing them with BPre using a

double-rail checker. The scheme can be used with any encoding, assuming a coder C is available. The advantage is to avoid any specific computation or logic optimization of the code prediction logic; this task is left to the synthesis tool.

Pairs of error detection signals are generated for groups of parity bits, according to a functional partitioning. 17 error detection bits are stored in two complementary registers Detect0 and Detect1, containing only zeros (respectively ones) in fault-free conditions. Due to the AHB interface, the alarm code is read on 32 bits, but only the 17 less significant bits give actual error indications. The only constraint imposed to the placement and routing of the circuit is to place these two registers far from each other, in order to avoid simultaneous modifications in these two registers by a single laser shot. This allows us to distinguish errors directly induced in these registers from errors detected in functional registers or combinatorial logic. All detected errors are memorized in the Detect registers until a reset of the coprocessor. An alarm signal is asserted as soon as at least one error has been detected.

The two parts of the dual-rail checkers, the coders, the detection registers and the functional logic are all implemented as separated blocks in the circuit hierarchy, so that a hierarchical synthesis avoids suppressing the redundancy implemented at RT level. The final hierarchy of the protected circuit is illustrated in Figure 4.

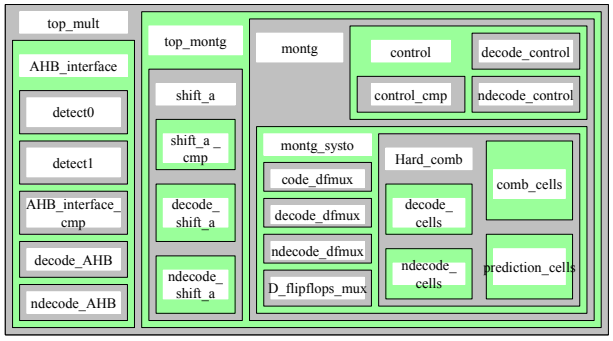

Figure 4: Global hierarchy of the protected circuit.

3.2. Overheads and level of protection

The two circuits have been implemented and manufactured in the ST HCMOS 130 nm process, with 6 metal layers. The complexity of the two cores after placement and routing (including clock trees and other amplification devices) is summarized in Table 1. The overhead is mainly due to the large systolic array, whose structure cannot be noticeably simplified in the parity prediction block due to its arithmetic functionality. For other types of circuits, the overhead could be smaller.

Table 1: Complexity of the two cores.

Circuit	Area (mm²)	# cells	# equivalent gates
Reference	0.434	26,589	55,751
Protected	0.827	46,084	116,796

In terms of protection level, the goal was not to fully protect the coprocessor against any type of fault but to achieve a reasonable trade-off between overheads and robustness. For errors occurring directly in the registers, the implemented protections achieve 100% detection of any single-bit error or any multiple-bit error with odd multiplicity. The detection of multiple-bit errors with even multiplicity depends on the repartition of the erroneous bits in the registers ; as soon as an odd number of bits is modified in one of the registers, an alarm is fired. The probability to detect errors in the systolic array depends on the number of erroneous outputs, and therefore on the gate-level structure. For the implemented circuit, the synthesis tool has voluntarily been used as in a standard flow, i.e. without any specific restriction on logic sharing or other structural optimizations. As a consequence, depending on the location of the fault(s) and on the logic state at the injection time, the erroneous outputs may be detected (odd multiplicity) or not (even multiplicity). Let us mention here that controlling the structure would not be very efficient, since a laser could illuminate a larger area than a single logic cone, thus potentially inducing multiple erroneous outputs with even multiplicity in spite of independent logic cones. On the opposite, we didn't constrain the placement and routing of the different components, in order to maximize the probability to simultaneously induce faults in multiple elements with a single laser shot, thus maximizing the probability that at least one of the elements detects the attack.

4. Experimental set-up

4.1. Gemalto equipment

Gemalto's laser fault injection platform is shown in Figure 5, with the board developed to hold the Montgomery demonstrator under the laser during the experiments. The platform is composed of a computer to organize both the fault injection and the driving of the device under test on an X-Y table to perform precise localization of the target in the circuit. An optional oscilloscope controls that the device under test receives commands and sends results. The laser itself is a pulsed Yag laser with a green output at 532 nm, an energy tunable from 0 to 100%, with the possibility to control the spot size.

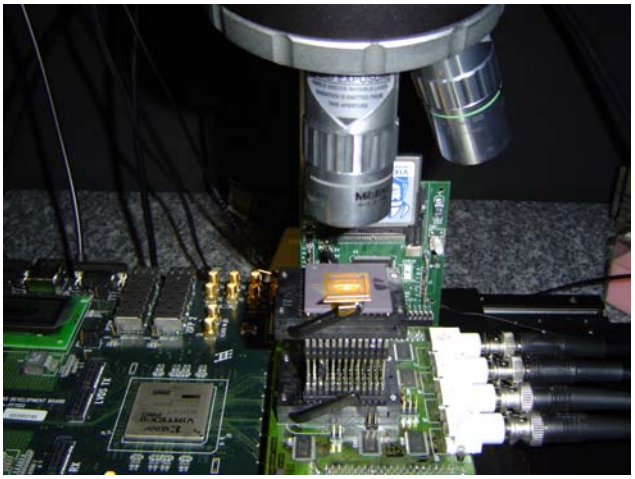

Figure 5: Laser platform with the Montgomery demonstrator test board.

4.2. Campaign specifications

In the case of the circuit under study, no area is a priori more critical than another from the application point of view. In fact, any undetected faulty multiplication may potentially be exploited to reduce security. It was therefore decided to scan the whole chip area and also to perform fault injections during the whole multiplication execution time. This corresponds to a black box approach.

In order to keep the experiment duration in a reasonable interval, some sampling has however been performed, i.e. injections have been performed at only 10 cycles during the execution, with a uniform repartition from cycle 0 to cycle 1350 (150 cycles separating two successive injection times). Also, the same input data (A, B, N) have been used for all experiments.

For the reference chip, the total area has been divided in 27 zones (9 X positions and 3 Y positions). For the protected chip, the total area has been divided in 45 zones (5 X positions and 9 Y positions). Each zone has an area of 147µm x 145µm and the beam is pointed to the center of the target zone. In order to study the determinism of the shot consequences, 5 separate shots have been performed at each spatial position for each injection time. This represents a total of 1350 shots on the reference circuit and 2250 shots on the protected circuit.

The energy used for all shots was "zero", that means the lowest possible energy level, corresponding to some "leakage beam".

Each shot was classified with respect to one of the following outcomes:

- Undetected wrong result: the output result was erroneous and no alarm was reported.
- No answer: the computation was blocked and no result was delivered.
- Detected wrong result: an alarm was reported (only in the case of the protected version).
- No effect: the computation ended with the correct result and no alarm.

To prevent latent fault effects, the circuit was reset and all input data were reloaded between any two experiments. Let us mention that the energy of the laser was tuned so that no hard error occurs.

5. Discussion of results

5.1. Classification results

The classification results are summarized in Table 2. For the two circuit versions, no shot was reported as having no functional effect. For the reference version, without specific protections, all shots led to either blocking the circuit or computing a wrong result. Among the 2250 shots on the protected version, only 6 led to an undetected wrong computation result and no blocking was recorded. This could be interpreted as a good result. However, the attack protocol was very simple and having 6 wrong results without alarm still demonstrates some vulnerability of the architecture. This is especially true taking into account that a "zero" energy level was used, in spite of a simple front side attack with 6 metal layers. These wrong results may not be exploitable from a cryptanalysis point of view in a real application context; however, the goal for a designer should be to avoid such vulnerabilities.

Table 2: Shot effects on the two cores.

Version	Undetected wrong results	No answer	Detected wrong results	No effect
Reference	89.58 %	10.42 %	--	0%
Protected	0.27 %	0 %	99.73 %	0%

Looking in detail to these 6 cases, it appears that they all correspond to a shot on the same zone, at five different cycles distributed from the beginning until almost the end of the multiplication execution. The exact reason of the sensitivity of this particular area has not yet been identified, since a lot of gates are within the corresponding zone; pinpointing the exact cause is a subject for further work. However, these results clearly indicate that multiple-bit errors with even parity should have been generated, since it is the only possibility to avoid detection when faults are induced in the systolic array or in the registers. It is currently not possible to say whether these error configurations are due to direct bit-flips in registers or to propagation of transients in combinatorial logic. A hypothesis may be some glitches induced on some flip-

flop reset inputs, since a few gates of the reset signal distribution tree are very close to the center of the spot at this position. It is also not possible to exactly know how many bits have been flipped. But anyway, these dangerous configurations have been quite easily obtained during the experiments and it is a strong indication for future hardening guidelines.

5.2. Determinism of the shot consequences

Looking more precisely to the recorded data, more detailed comments can be made and the first ones will be about determinism.

Position X	Position Y	Cycle	Class	Detect1	Detect0
1364.0	750.6	150	Alarm	FFFFCFFF	00003000
1364.0	750.6	150	Alarm	FFFFCFFF	00003000
1364.0	750.6	150	Alarm	FFFFEFFF	00001000
1364.0	750.6	150	Alarm	FFFFEFFF	00001000
1364.0	750.6	150	Undetected		
1654.0	895.6	1200	Alarm	FFFE34FF	0000C900
1654.0	895.6	1200	Alarm	FFFFB6FF	00004900
1654.0	895.6	1200	Alarm	FFFE94FF	00006900
1654.0	895.6	1200	Alarm	FFFFFEFF	00000100
1654.0	895.6	1200	Alarm	FFFFB6FF	00004900

Figure 6: Results obtained for two examples of shot sequences on the protected circuit.

As previously mentioned, each experiment has been repeated 5 times for the same spatial, temporal and energy parameters and with the same data processed by the circuit. Figure 6 shows two examples of such sequences of shots, clearly illustrating the lack of determinism. For each experiment, the contents of the detection registers Detect0 and Detect1 are shown if an alarm has been asserted. In the first sequence shown, four shots were detected but the fifth one remained undetected, although the conditions were exactly the same. Moreover, in the two first experiments, two different blocks generated an alarm (two bits are changed in the detection registers) while only one block detected the attack in the two other experiments. This may show a very high dependency of the sensitivity on the exact shot target, making any small difference in time and/or space, due to the precision tolerance values of the different equipments used in the set-up, very important. This is particularly surprising with a large spot size, as used during these experiments. But this may also be due to light diffusion phenomena in the 6 metal layers.

In the second sequence shown, the five shots were detected, but four different configurations are obtained in the detection registers. Among those, the first and the third trial led to asymmetric information. Asymmetric means that the dual rail encoding is not respected for all the bits. This corresponds to either direct bit-flips in one of the detection registers or to transients induced in only one part of a dual-rail decoder. Notice that the asymmetric bits are the same in these two cases. The symmetric bits correspond to error detections in the other blocks.

Table 3: Statistics on the determinism of shot effects.

Number of different error detection patterns	1	2	3	4	5
Number of injection times, over all spatial zones					
Min	0	0	0	0	0
Max	5	6	8	6	9
Average	0.56	1.42	2.69	2.44	2.89
Number of spatial zones, over all injection times					
Min	0	3	9	8	8
Max	4	9	14	14	16
Average	2.5	6.4	12	11	13
Global repartition over all spatial and temporal positions					
	5.56%	14.22%	26.89%	24.44%	28.89%

More global statistics are shown in Table 3. The top part of the table summarizes how many injection times have led to a given number of different patterns in the detection registers (these different patterns are between 1 and 5 since there are 5 trials for each experiment specification). The minimum and maximum numbers are obtained by comparing the repartition for the 45 possible zones. The average is obtained on all the 45 zones. A very deterministic effect should lead to only one detection pattern, the same for the five trials in each zone and no matter the injection time (i.e. the numbers 10-0-0-0-0 on the three lines in the table). As shown in the table, we are far from this situation, with a maximum of situations corresponding to 5 different patterns obtained for the 5 shots with identical specifications. Similarly, the bottom part of the table shows how many zones have led to a given number of different patterns, with respect to the 10 possible injection times. Here again, on an average, 13 out of 45 zones led to 5 different patterns for the 5 trials. The last line in Table 3 summarizes the global percentage, computed for all combinations of spatial and temporal specifications; almost 29% of the experiment

specifications led to 5 different error detection patterns, while less than 6% led to the same pattern for the 5 trials.

These results demonstrate a complex error propagation mechanism following the fault injection and this should be taken into account carefully when evaluating the robustness of some protection scheme. Real attack conditions are clearly far from the classical single bit-flip fault model.

5.3. Repartition of shots with respect to the number of asserted detection bits

As shown in Figure 6, more than one error detection bit is asserted in most cases after a single shot. This implies that the attack has been detected by several logic blocks in the circuit, confirming the generation of multiple effects in the logic networks. The repartition is illustrated in Figure 7, showing an average number of 6 blocks generating alarms due to the same shot, with a maximum of 13 blocks detecting simultaneously the attack! An average of 5.48 to 6.75 is obtained for the 10 different injection times, showing a small dependence on the computation step.

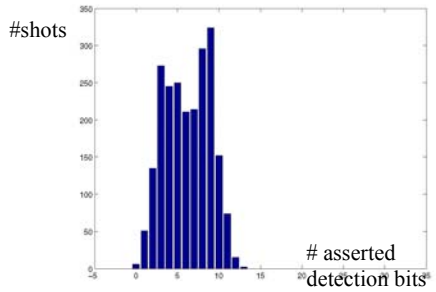

Figure 7: Repartition of shots with respect to the number of asserted detection bits.

6. Conclusion and perspectives

This paper presents original data about the effect of laser shots on a digital synchronous circuit including error detection mechanisms. These data are analyzed and compared with the effects observed on a similar circuit without protections.

The results show that at least with a given type of laser, the protection can leave open doors to an attacker in spite of a scheme that is not based on restrictive assumptions such as single-bit errors or unidirectional errors. The results give also insights into the fault models to be taken into account when designing a secured circuit and demonstrate that real attack conditions are clearly far from the classical single bit-flip fault model. On the opposite, large values of multiplicity must be taken into account (odd or even). Furthermore, the determinism of the effects is low, that makes still more difficult the analysis of the phenomena and the designer task.

The next fault injection campaign should focus on the zone of the protected circuit having led to undetected erroneous results. Scanning this zone with a smaller spot size and a smaller pitch would allow us to more precisely identify the exact reason of these undetected errors. Trying several sets of input data (A, B, N) may also lead to evaluate the impact of the processed data on the sensitivity.

Acknowledgements

The circuit prototype and the injections by laser have been partly supported by the French Ministry of Research, through the project RNRT DURACELL. The subsequent analyses carried out at TIMA are partly supported by the same Ministry, through the project ACI-SI MARS.

References

[1] V. Pouget, P. Fouillat, D. Lewis, H. Lapuyade, L. Sarger, F. M. Roche, S. Duzellier, R. Ecoffet, "An overview of the applications of a pulsed laser system for SEU testing", 6th IEEE International On-Line Testing Workshop, Palma de Mallorca, Spain, July 3-5, 2000, pp. 52-57

[2] A. K. Lenstra, "Memo On RSA Signature Generation In The Presence Of Faults", private communication (available from the author), September 28, 1996

[3] H. Bar-El, H. Choukri, D. Naccache, M. Tunstall, and C. Whelan, "The sorcerer's apprentice guide to fault attacks", Proceedings of the IEEE, vol. 94, no. 2, February 2006, pp. 370-382

[4] M.-L. Akkar, C. Giraud, "An implementation of DES and AES, secure against some attacks", CHES 2001, LNCS 2162, Springer-Verlag, 2001, pp. 309-318

[5] G. Bertoni, L. Breveglieri, I. Koren, P. Maistri, V. Piuri, "Error analysis and detection procedures for a hardware implementation of the Advanced Encryption Standard", IEEE Transactions on Computers, vol. 52, no. 4, April 2003, pp. 492-505

[6] Y. Monnet, M. Renaudin, R. Leveugle, N. Feyt, P. Moitrel, F. M'Buwa Nzenguet, "Practical evaluation of fault countermeasures on an asynchronous DES cryptoprocessor", 12th IEEE International On-Line Testing Symposium, Como, Italy, July 10-12, 2006, pp. 125-130

[7] D. Leroy, S. J. Piestrak, F. Monteiro, A. Dandache, S. Rossignol, P. Moitrel, "Characterizing laser-induced pulses in ICs: methodology and results", 12th IEEE International On-Line Testing Symposium, Como, Italy, July 10-12, 2006, pp. 11-16

[8] http://www.arm.com/products/solutions/AMBA_Spec.html

[9] M. Portolan, R. Leveugle, "A highly flexible hardened RTL processor core based on LEON2", IEEE Transactions on Nuclear Science, vol. 53, no. 4, part 1, August 2006, pp. 2069- 2075