Multi-spot Laser Fault Injection Setup: New Possibilities for Fault Injection Attacks

Brice Colombier, Paul Grandamme, Julien Vernay, Emilie Chanavat, Lilian Bossuet, Lucie de Laulanié, Bruno Chassagne

Multi-spot Laser Fault Injection Setup: New Possibilities for Fault Injection Attacks

Brice Colombier[1][0000−0002−6028−3028], Paul Grandamme[2], Julien Vernay[2], Émilie Chanavat[2], Lilian Bossuet[2][0000−0001−7964−3137], Lucie de Laulanié[3], and Bruno Chassagne[3]

[1] Univ. Grenoble Alpes, CNRS, Grenoble INP**, TIMA, 38000 Grenoble, France
brice.colombier@grenoble-inp.fr
[2] Univ. Lyon, UJM-Saint-Etienne, CNRS, Laboratoire Hubert Curien UMR 5516, F-42023, Saint-Étienne, France
{lilian.bossuet, julien.vernay, paul.grandamme}@univ-st-etienne.fr
emilie.chanavat@etu.univ-st-etienne.fr
[3] ALPhANOV Centre Technologique Optique et Lasers
Institut d'Optique d'Aquitaine, 33400 Talence, France
{lucie.bon, bruno.chassagne}@alphanov.com

Abstract. Fault injection attacks rely on experimental techniques to inject one or several faults into a device during operation. Among these techniques, laser fault injection is known as a powerful one, thanks to its unmatched spatial and temporal precision. So far though, the overwhelming majority of published laser fault injection attacks were performed with only one laser spot. In this article, we present a new multi-spot laser fault injection setup. After a description of the optical system, we highlight its new capabilities against the limitations of existing single-spot laser fault injection setups. We then discuss some intrinsic limitations that this setup has, making it not equivalent to running multiple single-spot setups simultaneously on the same target. We then provide experimental evidence of faults performed with two and four spots which are unfeasible with a single-spot laser fault injection setup. This paves the way for new fault attacks on security and cryptography algorithms that exploit this new type of fault.

Keywords: fault attacks · laser fault injection · multi-spot

1 Introduction

Faults induced in electronic systems by natural events, such as radiations, had been well studied for several decades by research in the safety domain. However, it was not until the article by Boneh *et al.* in 1997 that their importance with regards to security was acknowledged [7]. In that article, authors show how to take advantage of hardware faults to break cryptography algorithms. Since then,

** Institute of Engineering Univ. Grenoble Alpes

fault injection attacks have become a field of research in their own right [5]. In this setting, faults are injected *intentionally* to carry out the attack.

In order to induce a fault in an electronic system, an attacker has several tools to choose from [4]. We refer to the first category as *global* fault injection techniques: it is not possible to target a specific element of the system under attack. Among those techniques, we find voltages glitches [3] clock glitches [2] or heating [14]. The second category of fault injection techniques are *local*: they allow an attacker to target a specific feature of the device under attack. These techniques usually exploit radiations, either electromagnetic [16], optical [23] or in the form of X-rays [1].

In this article, we focus on optical fault injection. In particular, we deal with multi-spot laser fault injection setups. Compared with existing single-spot setups, multi-spot setups have the ability to inject multiple faults. This allows new types of faults to be performed, which are out of reach of single-spot setups, effectively extending the possible fault model. However, multi-spot laser fault injection setups also have some intrinsic limitations, due to the physical arrangement of optical elements. This constraint must be taken into account in the fault model. Finally, this extended fault model could be exploited to mount new attacks on security algorithms.

1.1 Contributions

This article makes the following contributions:

- We describe the different components of the optical apparatus used by the multi-spot laser fault injection setup,
- We show the possibilities of this new setup when compared to a single-spot laser fault injection setup,
- We highlight the limitations of a multi-spot laser fault injection setup, showing how mechanical and optical constraints lead to the fact that a multi-spot laser fault injection setup is not equivalent to multiple single-spot laser fault injection setups,
- We verify the capabilities of the setup by performing fault injection on characterisation codes. We experimentally perform two example faults, involving two and four laser spots respectively, that are impossible to achieve with a single-spot laser fault injection setup.

1.2 Outline

This article is organised as follows. Section 2 provides an overview of related work on laser fault injection. Section 3 describes the limitations of a single-spot laser fault injection setup with respect to the data corruption fault model. Section 4 presents the multi-spot laser fault injection, its capabilities as well as its intrinsic limitations. Section 5 provides experimental evidence for two new faults that can only be performed with a multi-spot laser fault injection setup. Finally, we conclude the article in Section 6.

2 Related work

Laser fault injection was first described in the context of hardware security by Skorobogatov and Anderson [23]. However, the action of photons on silicon devices was already known before. It was exploited to simulate the effect of ion beams and evaluate the reliability of integrated circuits [8].

As detailed in [20], when a laser shot passes through silicon, electron-hole pairs are created. If an electric field exists in the region, then these charges drift in opposite directions, inducing an electric current. This in turn may have an effect on the transistors, depending on their logic state before the laser shot. The exact sensitive areas of the transistors, which depend on the data handled, are detailed in [20]. Another important point is that, for the laser beam to penetrate deep enough in silicon and reach the active areas of the transistors, its wavelength must be in the infrared region, where silicon is transparent. Thus an infrared laser whose wavelength is in the micrometer range is commonly used for this purpose [11, 17, 20].

While access to the die is granted in the context of wafer-level testing, this is not the case for physical attacks. Thus the device under attack must first be decapsulated [5]. This can be done by chemical and mechanical means to dissolve the package and provide physical access to the die. An optional step of mechanical polishing can also be taken to thin the die, reducing absorption of the laser beam before it reaches the active areas of the transistors.

Pioneer work in laser fault injection was carried out on integrated circuits manufactured at micrometer-scale technology nodes. In [23], the target is a 6-transistor SRAM cell that has 20 µm on each side. This is of the same order of magnitude as the size of the laser spot used in this work, which had a diameter of 10 µm approximately. As technology nodes shrunk, the ability to perform precise laser fault injection was questioned. However, later work performed at the 90, 45 and 28 nm technology nodes showed that single bit faults are still within reach, by fine tuning the laser power [12, 21]. The correlation between the number of faulty bits and the laser power was explicitly established in [12]. A complex System-on-Chip was eventually attacked with this technique [24] and single-bit faults were observed in this case as well. Therefore, even though the features at a given technology nodes are far smaller than the laser spot size, laser fault injection remains a technique of choice for precise fault injection attacks.

Although the effect of multiple faults performed by laser fault injection was modeled at the register-transfer level in [19], no experiments were performed in this work. There are very few articles in the literature that claim to perform an attack using a multi-spot laser fault injection setup [6, 22, 25]. The first one [22] performs the same fault on two branches of an AES hardware implementation on an FPGA protected by redundancy, so the fault cannot be corrected. However, as noted by the authors, the attack relies on a very precise placement of the target elements, making it hard to reproduce on a real target design. The second one gives an overview of two certification processes followed by secure products, and describes the various tools which are used to perform the security evaluation [6]. The multi-spot laser fault injection setup is said to be capable of defeating

protected implementations, by shining one laser spot on the target while others are used to disable the hardware redundancy and cross-check verification. No further practical details were provided though. Finally, in [25], even though a two-spot setup is used, the fault models considered are described at the software level. Therefore, a lot of faults obtained cannot be explained and are referred to as "Fatal Errors": for instance, the target chip is not responding. Other valid faults are mostly classified as multiple instructions skip. Eventually, some faults are still left unexplained.

In this article, we chose instead to characterize the possibilities of a multi-spot laser fault injection setup with fault models that are fully explained and reproducible. For this reason, we focus on fault models that deal with data corruption in basic memory elements.

2.1 Fault model considered

We restrict our study here to works where clear evidence of data corruption by laser fault injection has been produced, either in the form of bit-set, bit-resets or bit-flips. In this regard, while previous work focused on memory elements like SRAM cells [20, 21] or D-flip flops [10] a recent line of work deals with NOR Flash memory architecture instead [11, 13, 15, 17]. The associated fault model is single or dual-bit bit-set. The photoelectric effect is still the root cause of the fault, and its effect on the NOR Flash architecture is detailed in [11, 17].

In other works, the occurrence of multi-bit faults was dependent on the physical layout of memory elements: a matrix of D flip-flops in a custom ASIC design in [12] and processor registers in [24]. Conversely, when performing laser fault injection in NOR Flash memory, it is not the individual memory elements that are faulty but the read-out circuitry. Data stored in the Flash memory remains unaffected by the fault. This makes the laser positioning much easier, since the bit-lines of the read-out circuitry are shared among memory bits. More precisely, bits of index i share the j^{th} bit-line such that $i \equiv j \bmod n$ where n is the width of data read from the Flash memory, usually 32 bits. Thus, traversing the memory lengthwise allows to fault the individual bits and their index is directly related to the position of the laser spot in the memory Flash length, as depicted in Figure 1. On our target device, which we will describe in more details in Section 4, the Flash memory has a length of 1500 µm, so individual bits can be targeted by making steps as large as $1500/32 \simeq 45$ µm. This is feasible manually with the joystick provided with most laser fault injection stations. We insist that we obtain the same perfect repeatability observed in [11, 17].

As experimentally demonstrated in [17], the results can be easily ported to a different target as long as it comes with NOR Flash memory. This is actually a very common feature in embedded systems where NOR Flash memory is used as EEPROM[4] to store the configuration of the microcontroller. For all these reasons, we chose to use this fault model to illustrate the possibilities of the multi-spot laser fault injection setup.

[4] Electrically-Erasable Programmable Read-Only Memory

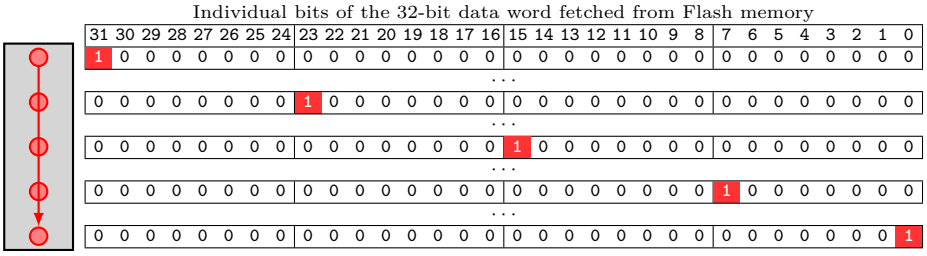


Fig. 1: Effect of the laser spot moving lengthwise over the Flash memory on all-zero 32-bit data (`0x00000000`) fetched from the Flash memory

3 Limitations of single-spot laser fault injection setups

Before introducing the multi-spot laser fault injection setup, it is important to identify the limitations of existing single-spot laser fault injection setups. To this end, we start by reviewing existing fault models for data corruption and identify two limitations in the way they are currently described.

3.1 Existing fault models for data corruption

We place ourselves here in the framework of fault injection attacks targeting data corruption. This choice brings us to a low level of abstraction, where we deal directly with data fetched from the Flash memory. If such data are instructions, then this could for example lead to the processor executing corrupted instructions, inducing another fault model, such as instruction skip, which is described at a higher level of abstraction. However, here, we consider the lowest possible level of abstraction to remain as general as possible.

The fault models dealing with data corruption capture quite well three aspects of the fault. The first one is the *direction* of the fault. Since we are dealing with binary data here, the different directions of the fault are: set (the data is forced to 1), reset (the data is forced to 0) or flip (the data is flipped from 0 to 1 or from 1 to 0).

The second one is the *cardinality* of the fault, that is, how many bits are affected by the fault. For instance the fault can have the following cardinalities: single-bit (one bit is faulty), multi-bit (multiple bits are faulty) or byte (eight bits are faulty).

The third one is the *repeatability* of the fault, that is, what is the probability that the fault occurs given a set of experimental fault injection parameters.

These existing characteristics of the data corruption fault models fail to capture two features of the fault, which are especially significant for the multi-spot laser fault injection setup. The first one is the contiguity of the fault and the second one is the time dimension of the fault model.

3.2 Contiguity

When a multi-bit fault model is considered, one aspect that is not taken into account is whether the faulty bits are contiguous or not.

When performing laser fault injection, the charges induced in silicon follow a Gaussian distribution [9]. The spread of this distribution depends on the laser power and is usually characterised by the "full-width at half-maximum" (FWHM) value [12, 24]. If the power is high, the area in which the charge density is high enough can be sufficiently large to encompass multiple transistors and induce a fault on multiple bits [12].

Based on this observation, one could argue that non-contiguous bits could be targeted by having multiple zones in the laser beam where the power is high enough. To achieve this, an SLM (Spatial Light Modulator), a DMD (Digital Micromirror Device), or a DOE (Diffractive Optical Element) can be used, that allows to split the incoming laser beam into multiple laser beams to target the device under attack. While these solutions may seem attractive, since they require only one laser source, the optical elements involved are complex and expensive. Moreover, the laser spots which are eventually focused on the device under attack are not fully independent, either spatially or temporally. In addition, the initial power is split among the beams, is dependent on their final shape and is hard to control. Therefore, these solutions make it very challenging to perform laser fault injection on non-contiguous bits with a single laser source in a controlled manner.

Another possibility to fault non-contiguous bits is to exploit the layout of target elements. For instance, if memory elements are organised in a grid shape, then injecting a fault on one side of the grid could lead to fault non-contiguous bits of the data. However, in this case, the fault model is layout-dependent, which obviously incurs a loss of generality.

Therefore, the first limitation of a single-spot laser fault injection setup is its inability to inject non-contiguous faults in general, as summarised in Figure 2.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

(a) Contiguous multi-bit fault: feasible with a single-spot setup

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0

(b) Non-contiguous multi-bit fault: not feasible with a single-spot setup, but feasible with a multi-spot setup

Fig. 2: Feasibility of contiguous and non-contiguous multi-bit faults with a single-spot laser fault injection setup

3.3 Time dimension

Another aspect which is not captured by existing fault models is the ability to perform two faults at different locations, but close in time. Indeed, with a single-spot laser fault injection setup, doing so requires to turn the laser off, move the target and turn the laser on again. On some setups, the objective lenses move while the target remains fixed, but the reasoning is identical. Indeed, the mechanical system can only be operated so fast. This should be contrasted with the clock frequencies at which the usual targets are operating, ranging from tens of megahertz to a few gigahertz [24].

If the time interval between the two intended faults is too small, then it is simply not possible to perform this type of fault with a single-spot laser fault injection setup. Let Δ_t be the time interval between the two faults, $v_{\max}$ the maximum linear speed of the mechanical setup and d_{targets} the distance between the two target features on the die. Then, for this type of fault to be feasible, we need the relation given in Equation (1) to hold.

$$\Delta_t > \frac{d_{\text{targets}}}{v_{\max}} \tag{1}$$

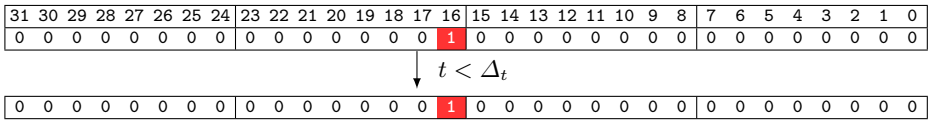
To simplify, we consider that the mechanical system always operates at full speed. In reality, the acceleration and deceleration phases are often sinusoidal to prevent abrupt changes in speed that could misalign the elements. With a realistic maximum linear speed of 20 mm/s and assuming that the features are distant of 10 % of a die that has 2 mm on each side, then the minimum time interval $\Delta_{t_{\min}}$ between the two faults is given in Equation (2).

$$\Delta_{t_{\min}} = \frac{d_{\text{targets}}}{v_{\max}} = \frac{2 \times \frac{10}{100}}{20} = 0.01\,\text{s} \tag{2}$$

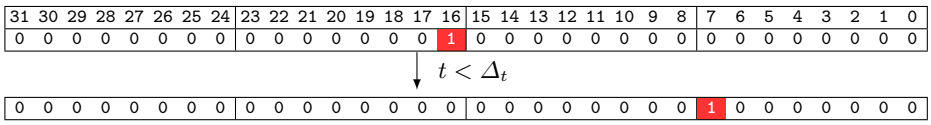
Considering a rather slow device running at only 10 MHz, that is, with a clock period of 100 ns, then $\Delta_{t_{\min}}$ is equal to 10^5 clock periods. This imposes a very hard constraint on the time interval between target instructions in a program if an attacker wants to perform multiple faults during its execution. These considerations are summarised in Figure 3.

Another critical aspect of having to move the target between two faults is the difficulty to synchronise these two faults together. Indeed, while the laser shots are very precise and synchronised with a trigger sent to the laser sources, the mechanical system cannot be synchronised precisely, adding a non-deterministic delay before the positioning is correct and the second laser shot can be made. Therefore, synchronising the two faults requires two triggers. This adds another constraint to the attack scenario.

As we will show in the next section, a multi-spot laser fault injection setup frees us from these constraints. It allows to perform multiple faults that are arbitrarily close in time without requiring multiple triggers.



(a) Consecutive faults on the same bit: feasible with a single-spot setup



(b) Consecutive faults on different bits: not feasible with a single-spot setup, but feasible with a multi-spot setup

Fig. 3: Feasibility of consecutive faults with a single-spot laser fault injection setup on a 16-bit data word

4 Four-spot laser fault injection setup

4.1 Setup description

Figure 4 shows the four-spot laser fault injection setup[5] used in the experiments.

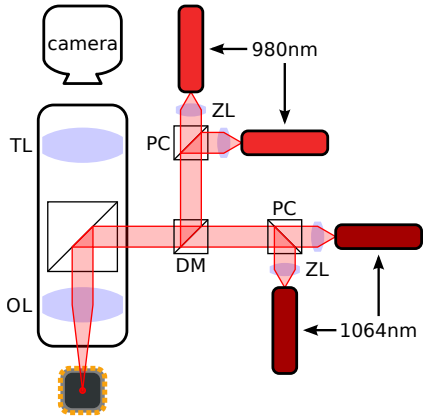


Fig. 4: Schematic of the four-spot laser fault injection setup (DM: dichroic mirror, PC: polarization beam splitter cube, OL: objective lens, TL: tube lens, ZL: zoom lenses).

Four monomode laser sources are integrated, two with a wavelength of 980 nm and two with a wavelength of 1064 nm. Monomode sources can be focused to

[5] QLMS by ALPhANOV : `https://www.alphanov.com/actualites/alphanov-con cu-un-banc-laser-quatre-spots-pour-linjection-de-fautes-sur-circuits`

smaller spots than multimode ones, allowing smaller features on the die to be targeted. The two laser sources of same wavelength are linearly polarized but perpendicular. They are combined by the polarization beam splitter cubes (PC) which are reflective for one direction of polarization and transmissive for the other one. The dichroic mirror then spectrally combines the laser beams of different wavelengths, reflecting the beam at 980 nm since it is reflective for this wavelength and tranmitting the beam at 1064 nm since it is transmissive for this wavelength. These are eventually focused on the die through the same objective lens (OL). Different objective lenses are available, namely x2.5, x20 and x50.

4.2 Capabilities

Each laser source is independent and can be moved across the focal plane in the optical field of view of the objective lens, allowing laser spots on the die to be positioned independently. Moreover, each laser source is triggered independently, allowing faults to be as close in time as required by the target application. The trigger signal may also be shared between multiple laser sources to perform simultaneous faults on distinct target elements.

4.3 Limitations

While the capabilities described above make it look like the four-spot laser fault injection setup is equivalent to four single-spot laser fault injection setups, this is in fact not the case. Indeed, since all laser beams must go through the same objective lens, the distance between the laser spots on the die is limited by the field of view of the objective. This distance between the spots depends on the magnification of the objective lens, which also affects the minimal laser spot diameter, as shown in Table 1.

Table 1: Field of view and minimal spot diameter for different objective lenses

Magnification	Field of view	Minimal spot diameter
x2.5	4 mm	25 µm
x20	500 µm	2.2 µm
x50	200 µm	1.3 µm

For instance, with a x20 magnification, the laser spots cannot be more than 500 µm apart from one another. Therefore, if the targets elements on the die are further apart than this limit, they cannot be targeted at the same time. Doing so would require to move the target, which as detailed above is unrealistic in most attack scenarios. In addition, with this magnification, the laser spot cannot have a diameter smaller than 2.2 µm. As mentioned before, this is not an obstacle when aiming for single-bit faults, since we can tune the laser power so that only a smaller area has a charge density high enough to cause a fault.

Another aspect relative to the laser spot positions is the fact that, when moved away from the center of the field of view, they gradually lose power, as shown in Figure 5. While barely visible for x2.5 and x20 objective lenses, this effect is very strong for the x50 objective lens. Indeed, in this setting, if a laser spot is positioned on the edge of the field of view, then almost no optical power reaches the die. This turned out not to be an issue in the following experiments, since we used the x20 objective lens only.

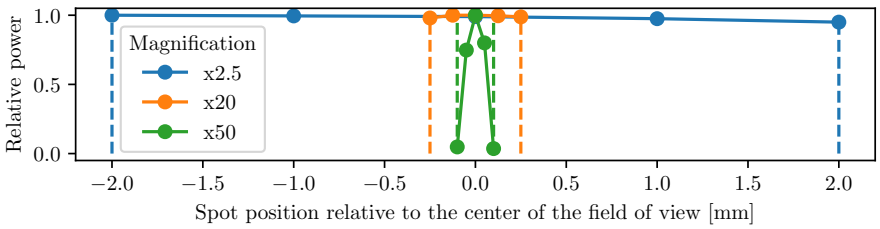
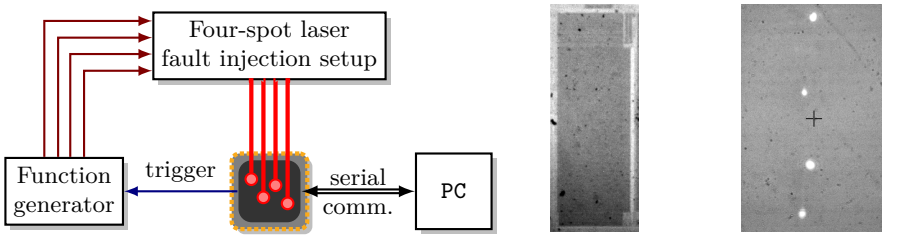


Fig. 5: Relative power of laser for different spot positions in the field of view of different objective lenses

5 Two examples of new possible faults

5.1 Experimental setup

Full experimental setup The hardware target communicates with a PC over a serial interface. It generates a trigger signal, sent to a function generator, which generates the four distinct control signals for the laser sources. This is shown in Figure 6a. A picture of the Flash memory area of the microcontroller is shown in Figure 6b while Figure 6c shows four laser spots over the Flash memory.



(a) Experimental setup (b) Flash memory (c) Four laser spots

Fig. 6: Four-spot laser fault injection in Flash memory

Hardware target The hardware target we perform the experiments on is a 32-bit microcontroller, integrated on a custom target board for the ChipWhisperer platform [18] to allow for backside access. The microcontroller embeds an ARM Cortex-M3 core and comes with 128 kB of integrated Flash memory. It runs at a frequency of 7.4 MHz, as dictated by the ChipWhisperer platform.

Laser fault injection setup parameters After characterisation, following the method detailed in [17], we set the laser power to 1.5 W to obtain single-bit faults on data fetched from Flash memory, with one laser spot only, a 980 nm laser source and the x20 objective lens. The duration of the laser pulse was set to 135 ns, which is the clock period of the microcontroller. We observed that, on this hardware target, the laser spot must be moved in steps of 45 µm to perform a transient fault on the individual bits of data fetched from the Flash memory.

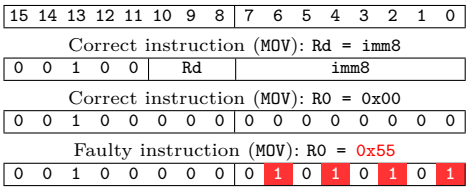
5.2 First characterisation code

The goal of this first code is to validate the possibility to perform simultaneous non-contiguous faults. For that, we target a `MOV` instruction that loads an 8-bit value in a register, as shown in Figure 7a where `0x00` is loaded in `R0`. We raise a trigger signal before the target instruction and lower it after, before reading back the content of the `R0` register.

This source code is compiled using the Thumb instruction set without any optimisation. Figure 7b shows how this instruction is encoded. We aim for the `imm8` part of the instruction and want to load `0x55` instead of `0x00`, to demonstrate the ability to perform four simultaneous non-contiguous bit-sets.

```
1  // Raise trigger signal
2  MOV R0, 0x00
3  // Lower trigger signal
4  // Read back R0
```

(a) Assembly source code



(b) Faulty data loading

Fig. 7: Characterisation code for four simultaneous faults on non-contiguous bits

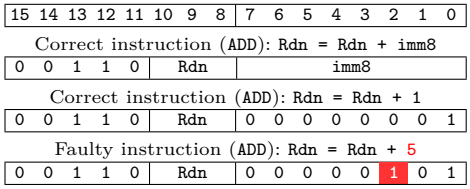
Experimental results We started the experiment with only one laser spot, with the experimental parameters given above. We gradually increased the delay up to 1113 ns where a single-bit fault was observed. We then positioned the three other spots with a distance of 90 µm between them, since there is a step of 45 µm between individual bits. We had to lower their individual power to approximately 750 mW, otherwise the chip crashed and was not responding anymore. Finally, we succeeded to store the value `0x55` in `R0`.

5.3 Second characterisation code

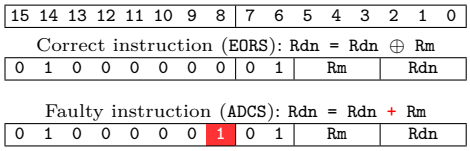
The second fault consists in targeting two instructions which are close, but perform the fault on different bits. To this end, we use the characterisation code shown in Figure 8a.

```
1  #define N_ITER 1000
2  void charac_func(void) {
3    volatile uint32_t ref_count = 0;
4    uint32_t results[2] = {0, 0};
5    uint32_t XOR, ADD = 0;
6    trigger_high();
7    for (volatile uint32_t iter = 1;
8         iter <= N_ITER;
9         iter++)
10   {
11     ref_count++;
12     XOR = iter ^ iter;
13     ADD = iter + iter;
14     results[1] += (XOR == ADD);
15   }
16   results[0] = N_ITER - ref_count;
17   trigger_low();
18   // Read back results
19 }
```

(a) C source code with target instructions pointed by arrows



(b) Faulty loop increment



(c) Faulty exclusive-OR operation

Fig. 8: Characterisation code for two faults close in time on different bits

Again here, we raise a trigger signal at the beginning of the execution (see Figure 8a, line 6) and lower it at the end (see Figure 8a, line 17)

We target the instructions associated with the following two operations:

- the increment of the loop counter. This compiles into an `ADD` instruction as shown in Figure 8b. We perform a fault injection on the `imm8` part of the instruction. More specifically, we modify the increment to make it N instead of 1. We assume a single-bit bit-set fault model, so N is of the form $2^i + 1$, where i is an integer between 1 and 7. This requires to perform a bit-set on the bit of index i. For example, the increment can be changed to 5 by performing a bit-set on the bit of index 2, as shown in Figure 8b.
- an exclusive-OR in the body of the loop. This compiles into an `EORS` instruction as shown in Figure 8c. We perform a fault injection on the opcode, turning the `EORS` instruction into an `ADCS` instruction, to perform an addition with carry instead. This requires to perform a bit-set on the 8^{th} bit.

The experimental results are stored in an array of two elements. The first one stores the difference between the original and the actual number of times the body of the *for* loop has been executed. The second one stores the number of times the exclusive-OR operation has been turned into an addition. This way, we isolate the two faults and are able to observe their respective influences.

Experimental results We performed different experiments by changing the increment of the loop counter to different values, while faulting the `EORS` instruction in the body of the *for* loop at the same time. As specified above, the hardware target sends only one trigger signal. From there on, in order for the fault injection to be successful, the main challenge is to find the correct parameters for the two control signals of the two laser sources. To this end, four parameters must be tuned on the function generator:

- the initial delay for the first laser source t_{init_1}. This is the delay between raising of the trigger signal and executing the first instruction to fault.
- the initial delay for the second laser source t_{init_2}. This is the delay between raising of the trigger signal and executing the second instruction to fault.
- the period t_{lasers} which is the time it takes to execute the body of the *for* loop once. Note that both control signals have the same period.
- the duty cycle α which defines how long every laser shot should last. Since we want each laser to fault one instruction per execution of the body of the *for* loop, the duty cycle must be set accordingly.

The first step is to tune the two initial delays. This is done by increasing these delays one after the other while monitoring the result values. As soon as one fault is observed, the initial delay is found. We obtain the following values: $t_{\text{init}_1} = 2070\,\text{ns}$ and $t_{\text{init}_2} = 3825\,\text{ns}$. The second step is to tune the period of the control signals. This is done by producing only two pulses and increasing the period until two faults are observed. We obtain the following value: $t_{\text{lasers}} = 5535\,\text{ns}$. This corresponds to 41 clock periods ($5535 = 41 \times 135$) given that our target has a clock period of 135 ns. Therefore, executing the body of the *for* loop takes 41 clock cycles. Finally, we set the duty cycle to $\alpha = \frac{1}{41} \simeq 2.4\,\%$ to target one clock cycle out of the 41 of the body of the *for* loop.

These four settings are shown in Figure 9, where the actual fault performed by each laser shot is shown as well. Using this settings, we were able to change the loop increment to 5 instead of 1 and alter the exclusive-OR operation in the body of the loop to turn it into an addition.

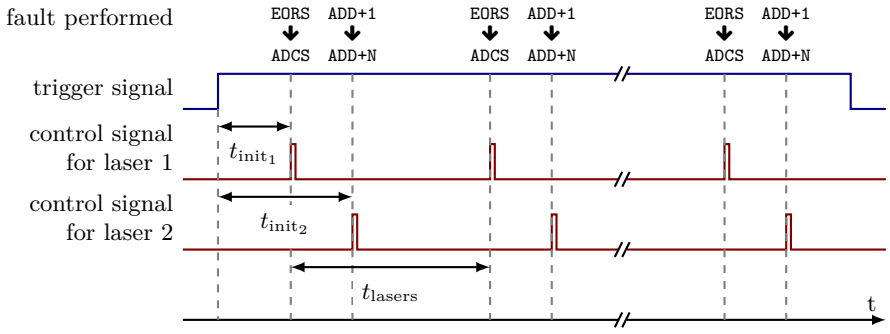


Fig. 9: Timing of signals used to control the fault injection setup

6 Conclusion

In this article, we highlighted the limitations of single-spot laser fault injection setups, which are not considered by previously considered fault models. We then presented a new four-spot laser fault injection setup that can overcome these limitations. With experiments on two characterisation codes, we showed that two new types of fault are feasible with this setup: four simultaneous non-contiguous faults and two faults very close in time on different bits. Having identified this extension of the data corruption fault model, feasible by laser fault injection, future works could focus on applying this to new attacks on security algorithms.

Acknowledgement

This work was carried out in the framework of the FUIAAP22 Project PILAS supported by Bpifrance. This work is supported by the French National Research Agency in the framework of the "Investissements d'avenir" program "ANR-15-IDEX-02" and the LabEx PERSYVAL "ANR-11-LABX-0025-01". This work is supported by INS2I in the framework of the PANTACOUR project.

The authors would also like to thank Jean-Max Dutertre from EMSE for providing them with a backside-opened device suitable for laser fault injection.

References

1. Anceau, S., Bleuet, P., Clédière, J., Maingault, L., Rainard, J.-L., and Tucoulou, R.: Nanofocused X-Ray Beam to Reprogram Secure Circuits. In: International Conference on Cryptographic Hardware and Embedded Systems, pp. 175–188 (2017)
2. Anderson, R.J., and Kuhn, M.G.: Low Cost Attacks on Tamper Resistant Devices. In: International Workshop on Security Protocols, pp. 125–136 (1997)
3. Aumüller, C., Bier, P., Fischer, W., Hofreiter, P., and Seifert, J.-P.: Fault Attacks on RSA with CRT: Concrete Results and Practical Countermeasures. In: CHES, pp. 260–275 (2002)
4. Bar-El, H., Choukri, H., Naccache, D., Tunstall, M., and Whelan, C.: The Sorcerer's Apprentice Guide to Fault Attacks. Proceedings of the IEEE (2006)
5. Barenghi, A., Breveglieri, L., Koren, I., and Naccache, D.: Fault Injection Attacks on Cryptographic Devices: Theory, Practice, and Countermeasures. Proceedings of the IEEE (2012)
6. Bhasin, S., Lomné, V., and Tobich, K.: An Industrial Outlook on Challenges of Hardware Security in Digital Economy. In: International Conference on Security, Privacy, and Applied Cryptography Engineering, pp. 1–9 (2017)
7. Boneh, D., DeMillo, R.A., and Lipton, R.J.: On the Importance of Checking Cryptographic Protocols for Faults. In: International Conference on the Theory and Application of Cryptographic Techniques, pp. 37–51 (1997)
8. Buchner, S., Kang, K., Stapor, W., Campbell, A., Knudson, A.R., Mcdonald, P.T., and Rivet, S.: Pulsed laser-induced SEU in integrated circuits: a practical method for hardness assurance testing. IEEE Transactions on Nuclear Science (1990)
9. Buchner, S., Knudson, A.R., Kang, K., and Campbell, A.: Charge collection from focussed picosecond laser pulses. IEEE Transactions on Nuclear Science (1988)

10. Champeix, C., Borrel, N., Dutertre, J.-M., Robisson, B., Lisart, M., and Sarafianos, A.: SEU sensitivity and modeling using pico-second pulsed laser stimulation of a D Flip-Flop in 40 nm CMOS technology. In: International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems, pp. 177–182 (2015)
11. Colombier, B., Menu, A., Dutertre, J.-M., Moëllic, P.-A., Rigaud, J.-B., and Danger, J.-L.: Laser-induced Single-bit Faults in Flash Memory: Instructions Corruption on a 32-bit Microcontroller. In: IEEE International Symposium on Hardware Oriented Security and Trust, pp. 1–10 (2019)
12. Dutertre, J.-M., Beroulle, V., Candelier, P., Castro, S.D., Faber, L.-B., Flottes, M.-L., Gendrier, P., Hély, D., Leveugle, R., Maistri, P., Natale, G.D., Papadimitriou, A., and Rouzeyre, B.: Laser Fault Injection at the CMOS 28 nm Technology Node: an Analysis of the Fault Model. In: FDTC, pp. 1–6 (2018)
13. Garb, K., and Obermaier, J.: Temporary Laser Fault Injection into Flash Memory: Calibration, Enhanced Attacks, and Countermeasures. In: International Symposium on On-Line Testing and Robust System Design, pp. 1–7 (2020)
14. Hutter, M., and Schmidt, J.: The Temperature Side Channel and Heating Fault Attacks. In: International Conference on Smart Card Research and Advanced Applications, pp. 219–235 (2013)
15. Kumar, D.S.V., Beckers, A., Balasch, J., Gierlichs, B., and Verbauwhede, I.: An In-Depth and Black-Box Characterization of the Effects of Laser Pulses on ATmega328P. In: International Conference on Smart Card Research and Advanced Applications, pp. 156–170 (2018)
16. Maurine, P.: Techniques for EM Fault Injection: Equipments and Experimental Results. In: FDTC, pp. 3–4 (2012)
17. Menu, A., Dutertre, J.-M., Rigaud, J.-B., Colombier, B., Moëllic, P.-A., and Danger, J.-L.: Single-bit Laser Fault Model in NOR Flash Memories: Analysis and Exploitation. In: Workshop on Fault Detection and Tolerance in Cryptography, pp. 41–48 (2020)
18. O'Flynn, C., and Chen, Z.: ChipWhisperer: An Open-Source Platform for Hardware Embedded Security Research. In: International Workshop on Constructive Side-Channel Analysis and Secure Design, pp. 243–260 (2014)
19. Papadimitriou, A., Hély, D., Beroulle, V., Maistri, P., and Leveugle, R.: A multiple fault injection methodology based on cone partitioning towards RTL modeling of laser attacks. In: Design, Automation & Test in Europe Conference & Exhibition, pp. 1–4 (2014)
20. Roscian, C., Sarafianos, A., Dutertre, J.-M., and Tria, A.: Fault Model Analysis of Laser-Induced Faults in SRAM Memory Cells. In: FDTC, pp. 89–98 (2013)
21. Selmke, B., Brummer, S., Heyszl, J., and Sigl, G.: Precise Laser Fault Injections into 90 nm and 45 nm SRAM-cells. In: International Conference on Smart Card Research and Advanced Applications, pp. 193–205 (2015)
22. Selmke, B., Heyszl, J., and Sigl, G.: Attack on a DFA Protected AES by Simultaneous Laser Fault Injections. In: FDTC, pp. 36–46 (2016)
23. Skorobogatov, S.P., and Anderson, R.J.: Optical Fault Induction Attacks. In: CHES, pp. 2–12 (2002)
24. Vasselle, A., Thiebeauld, H., Maouhoub, Q., Morisset, A., and Ermeneux, S.: Laser-Induced Fault Injection on Smartphone Bypassing the Secure Boot-Extended Version. IEEE Transactions on Computers (2020)
25. Werner, V., Maingault, L., and Potet, M.-L.: An End-to-End Approach for Multi-Fault Attack Vulnerability Assessment. In: Workshop on Fault Detection and Tolerance in Cryptography, pp. 10–17 (2020)