RESEARCH ARTICLE

Dependability evaluation of integrated circuits at design time against laser fault injection

Huiyun Li* and Hai Yuan

Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences, The Chinese University of Hong Kong, Xueyuan Avenue, Xili, Shenzhen 518055, China

ABSTRACT

Laser fault injection has been proved to be a useful tool for attacks on integrated circuits. Transistors hit by a pulse of photons causes them to conduct transiently, thereby introducing transient logic errors, such as register value modifications, memory dumping, and so on. Attackers can make use of this abnormal behavior and extract sensitive information that the devices try to protect. This paper demonstrates laser fault injection attacks on very-large-scale integration circuits in a semi-invasive way for the purpose of validating fault tolerant design and performance. Then, the paper presents a simulation methodology to evaluate the dependability of the integrated circuit design against laser fault injection attacks during design time. This simulation methodology involves exhaustively scanning the layout, incorporating the exposed cells into a circuit simulator, and examining the response of the circuit in detail. Experiments conducted on the same test chip spot the same vulnerabilities, thus indicating the validity of the proposed simulation methodology. Copyright © 2011 John Wiley & Sons, Ltd.

KEYWORDS

laser fault injection; dependability; evaluation; integrated circuit design

*Correspondence

Huiyun Li, Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences, The Chinese University of Hong Kong, Xueyuan Avenue, Xili, Shenzhen 518055, China.

E-mail: hy.li@siat.ac.cn

1. INTRODUCTION

Dependable integrated circuit (IC) design against harsh environments is an important research and development area. Dependable ICs should endure attacks that inject faults into the devices to cause exploitable abnormal behavior. The abnormal behavior may be a data error, setting part of the key to a known value or a missed conditional jump reducing the number of rounds in a block cipher. A glitch attack was the most widely known fault injection technique [1,2]. In a glitch attack, the attacker deliberately generates a malfunction that causes one or more flip-flops to transition into a wrong state. The aim is usually to replace a single critical machine instruction with an almost arbitrary one. Glitches can also aid in corrupting data values as information is transferred between registers and memory.

A supply voltage glitch during execution may cause a processor to misinterpret or skip instructions. An external clock glitch may cause data misread, when the circuit tries to read a value from the data bus before the memory had time to latch out the asked value. An external clock glitch may also cause an instruction miss when the circuit starts executing instruction \( n + 1 \) before the microprocessor finished executing instruction \( n \) [2]. In temperature variations using an alcoholic cooler until the chip temperature cools down to a value where write operations work but read operations do not, or the other way around, a number of attacks can be mounted. Many chips nowadays are designed to detect these glitch attacks; therefore, detecting these glitch attacks is not the concern of this paper.

Laser fault injection introduced by Skorobogatov and Anderson [3] appears to be a powerful and dangerous attack. It involves illumination of a target transistor, which causes the transistor to conduct transiently, thereby introducing a transient logic error. Such attacks are practical as they do not require the expensive equipment that is needed in invasive attacks.\(^1\) This threat has become increasingly relevant as transistor dimensions and supply

\(^1\)Invasive attacks require decapsulation and deprocessing to get direct access to the internal components of the device.
voltages are constantly scaling down. In deep submicron technologies, it is easier to introduce and propagate transient voltage disturbances as the capacitance associated with individual circuit nodes is very small, and large voltage disturbances can be produced from relatively small amounts of ionized charge. Also, because of the high speed of deep submicron circuits, the voltage disturbances can propagate more easily.

To keep ICs secure and dependable against laser fault induction attacks, various ideas have been proposed [6,24]. To evaluate these research efforts, a design-time \textit{dependability evaluation} methodology is proposed to exhaustively examine the response of the processors under optical illumination by simulation so as to assess their dependability level against laser fault injection attacks at design time.

The rest of the paper is organized as follows. In Section 2, we introduce the physical mechanism of laser radiation, ionization, and charge absorption. In Section 3, we present our simulation methodology that includes layout scanning, exposed node list extraction, and circuit simulation that incorporates transient voltage supplies to these exposed nodes. In Section 4, we demonstrate simulation results on our test chip from which vulnerability is successfully identified and verified by experiment results. Section 5 presents a discussion of defense technologies followed by a brief conclusion in Section 6.

\section{2. BACKGROUND}

Research on laser fault injection has been conducted because semiconductor devices were invented, which were found to be sensitive to ionizing radiation in space, caused by protons, neutrons, alpha particles, or other heavy ions [4]. Pulsed lasers were then used to simulate ionizing radiation on semiconductors [5]. Laser illumination may cause various consequences: no observable effect, a transient disruption of circuit operation, a change of logic state, or even permanent damage to the device under test [6].

\subsection{2.1. Ionization and charge collection}

Laser ionization and absorption has long been known as a fundamental band-to-band absorption process, where a pulsed laser with photon energy greater than the band gap of the semiconductor material may excite carriers from the valence to the conduction band [7] and may produce electron-hole pairs within semiconductor materials. Each absorbed photon is assumed to produce a single electron-hole pair, and the light is absorbed exponentially with depth $x$. Beer’s Law describes the laser intensity function as $I = I_0 e^{-ax}$, where the absorption coefficient $\alpha$ is strongly dependent on the wavelength of the laser light $\lambda$ and has been assumed to be constant for old device technologies. However, the assumption of linear absorption is no longer valid for new silicon-based technologies and most GaAs technologies for a number of reasons. First, the absorption coefficient $\alpha$ varies with temperature. For silicon, $\alpha$ approximately doubles at 125°C compared with its value at room temperature. Secondly, at high doping levels, the presence of a large number of impurities reduces the energy gap and hence increases the absorption coefficient [7]. Thirdly, when pulsed lasers are focused to small spots, the resulting high-power densities may cause additional absorption mechanisms such as two-photon absorption, which involves simultaneous absorption of two photons and thus a highly nonlinear increase in the absorption [7]. Furthermore, free-carrier absorption may occur and will not produce ionization but increases the energy of carriers within conduction or valence bands.

If the excited charge amount reaches the critical charge $Q_{\text{crit}}$ (the charge necessary to convert a binary bit “1” to “0” or vice versa), a single-event upset (SEU) occurs. Device vulnerability is determined by its threshold linear energy transfer (LET). The threshold LET (LET$_{\text{th}}$) is defined as the minimum LET required to produce a voltage change ($\Delta V$) sufficient for an SEU, then mathematically:

$$\text{LET}_{\text{th}} \times \Delta V = \frac{Q_{\text{crit}}}{C}$$

where $C$ is the capacitance of the struck node.

\subsection{2.2. Substrate and well}

In a microelectronic device under laser illumination, the most sensitive regions are usually the reverse-biased pn junctions. The high field present in a reverse-biased junction depletion region can efficiently collect the ionized charge through drift processes, leading to a transient current at the junction contact. Whether the junction is located inside a well or in the substrate has considerable effect on the consequence. Figure 1 shows a cross-section of a complementary metal–oxide–semiconductor (CMOS) inverter in a p-substrate with n-well process. There are other substrate and well types, including the following:

- p-Substrate (n-MOS) + n-well (p-MOS),
- p-Substrate + twin-well (p-MOS in n-well and n-MOS in p-well),
- n-Substrate (p-MOS) + p-well (n-MOS),
- n-Substrate + twin-well (p-MOS in n-well and n-MOS in p-well).

With the development of shrinking technologies, inside-the-well strikes become more and more interesting because shunt and bipolar effects can occur in multilayer structures [8]. Electronic holes deposited in the p-well are collected at the p-substrate contact, raising the
well potential and leading to injection of electrons by the source. This results in the turn-on of the horizontal n-source/p-well/n-drain (emitter/base/collector) parasitic bipolar.

Dodd *et al.* [8] studied the inside-the-well strikes under the gate-length scaling trend, for both p-substrate and n-substrate technologies (Sandia 2, 1, and 0.5 µm).

**F2** Figure 2 shows the simulated SEU threshold under the gate-length scaling trend of off transistors fabricated on an n-substrate, illustrating that the upset threshold of the inside-the-well strikes decreases at a much faster rate than that of outside-the-well strikes. Figure 3 displays the scaling trend of off transistors fabricated on a p-substrate. A similar trend appears except that the inside-the-well (p-MOS) strike starts out much harder (much higher LET threshold than the n-MOS counterpart). This is due to a weaker pnp bipolar effect in the p-well than the npn bipolar effect in the n-well. For identical structures, a pnp bipolar will have lower current gain (~1/3) than an equivalent npn due to the lower mobility of holes compared with electrons.

According to the trend analyzed in [8], a rule of thumb is

- For p-substrate, either p-substrate + n-well or p-substrate + twin-well: n-MOS is easier to switch on.
- For n-substrate, either n-substrate + p-well or n-substrate + twin-well: above 1 µm technology node, p-MOS is easier to switch on; below 1 µm technology, device simulation or experiment is required to determine the minimum upset LET for n-MOS and p-MOS, respectively, before the proposed simulation methodology is applied on a large scale IC.

The rule of thumb does not hold for the silicon-on-insulator technologies, but it is not discussed in this paper as almost all cryptographic ICs nowadays are designed and fabricated in bulk silicon technologies.
2.3. Metal shielding effect

Metal on the top of the sensitive junctions prevents the light from penetrating these regions directly so that has to be taken into consideration for the laser fault injection. The metal shielding reduces the average incident energy in proportion to the surface metallization [9]:

\[ P^m(x) = P_x(x)(1-K_m) \]  

(2)

where \( P_x(x) \) is the incident energy without metal shielding effect, \( P^m(x) \) is the incident energy with metal shielding effect, \( K_m = S_m/S \), \( S \) is the total top surface area under illumination, whereas \( S_m \) is the metallization area within.

A way to bypass metal shielding is to attack the chip from the back, if the target device allows this.

2.4. Classes of attackers

Attackers of IBM cryptographic products are classified into three types according to their abilities and attack strengths [10]. Following this classification, we categorize the laser fault induction attackers into three types according to their knowledge about the system and the resolution that their laser scan equipment allows:

**Type I** (not knowing layout, targeting many transistors).
They are outsiders with moderately sophisticated tools. They do not have detailed knowledge of the layout and can only perform moderately low-resolution scans of the chip, targeting a group of neighboring transistors.

**Type II** (not knowing layout, targeting a single transistor).
They are outsiders with sophisticated tools. They do not have detailed knowledge of the layout but can perform high-resolution scans of the chip, targeting individual transistors in order to determine what faults can be injected.

**Type III** (knowing layout, targeting a single transistor).
They are knowledgeable insiders, having detailed information of the layout of the chip under attack and information about the program code. They also have access to highly sophisticated tools such as a probing station with a high-resolution focused laser allowing any single transistor to be targeted.

Type I attackers stand for the largest group of potential attackers, because the costs for training, intelligence, and equipment are relative low. The ease of Type I attacks indicates that they are the most dangerous and thus are the focus of this work.

Type II and III attackers on the other hand can conduct an attack on any transistor node during an application program execution, knowing or without knowing its specified functionality. The requirement of large investment and detailed internal knowledge prevent most attackers from falling into these categories. However, such attackers have higher capability to manipulate the circuit, so more defensive effort is required from chip designers.

2.5. Modeling laser fault injection

Fault injection simulation for radiation effects can be conducted at a number of different levels, from physical device models through to digital abstractions.

2.5.1. Device modeling.

Device simulation consisted of 1-D drift-diffusion (DD) models is discussed in [12]. In a DD model, current equations are derived from the Boltzmann transport equation considering a steady state situation and numerical approximations for a 1-D geometry. These equations are discretized and solved on a mesh using finite-difference or finite-element techniques [13]. However, the models become inappropriate in treating the effects for small-geometry devices, such as velocity overshoot, carrier heating, and quasi-ballistic transport [14]. Nevertheless, DD models are highly evolved and, relatively speaking, not terribly computationally intensive, except in the case of 3-D models. DD models remain as the mainstream simulation tools, even for deep submicron devices.

The alternative device modeling strategy is based on hydrodynamics and energy balance (EB). It has fewer assumptions [14] but is more computationally intensive, based on five or six equations of state rather than the three used in the DD method.

Another method is to use stochastic solution methods (Monte Carlo) for device simulation, which involve the simulation of particle trajectories rather than the direct solution of partial differential equations. It describes carrier transport on a fundamental, microscopic scale using classical equations of motion (e.g., Newton’s first law). The motion of individual carriers is followed as they drift in fields and interact with scattering centers until statistical significance is achieved. Compared with the previous two approaches, Monte Carlo simulation makes the fewest assumptions and approximations [14], however, it has a very high computational intensity as the trajectories of many thousands of particles must be tracked to attain meaningful statistics.

The 1-D device models based on DD equations for carrier densities and models based on hydrodynamic and EB have evolved to 2-D and 3-D device modeling approaches. Many charge collection and SEU studies have been performed using these models. An early comparison of 2-D and 3-D charge-collection simulations showed that although the transient responses were qualitatively similar, quantitative differences existed in both the magnitude of the current response and the time scale over which collection was observed [15]. The comparison implies that 2-D simulations can provide basic insight, whereas 3-D simulations become
2.5.2. Circuit simulations.

Although full 3-D device simulators were first reported in the literature in the early 1980s [16], only in the last few years have full 3-D device simulators become commercially available [6]. Even optimized for high-end workstations, a fairly large 3-D device simulation can still take a few hours. Even 2-D device modeling is too computationally expensive for simulating responses of a large circuit to a laser fault injection. Therefore, in order to exhaustively examine the effect of laser fault injection on a large circuit, we need to relate the collection of charges in individual device junctions to the charges in the circuit currents and voltages. A common circuit model for charge collection at a junction due to direct funneling or diffusion is a double-exponential, time-dependent current pulse [17], with a typical rise time on the order of 10s of picoseconds and a fall time on the order of 200 to 300 ps [18]. The actual magnitude and time profile of the current model depend on material parameters, ion species, ion energy, device dimensions, and hit location relative to the junction. If the time profile (or the shape) of the collection current pulse is not important to the circuit response to the hit, then analytical current models can usually adequately describe the induced current pulse. If, however, the time profile is critical to the circuit response, more accurate models for the current pulse are necessary, such as those derived from a device simulation. In a laser fault injection attack introduced in [3], the shape of the collection current is not important to the circuit response to the attack, and a piecewise linear pulse can even be used to represent the induced current pulse for the purpose of simplicity.

2.5.3. Mixed device/circuit simulations.

Recently, the simultaneous solution of device and circuit equations has been increasingly used. With this technique, known as mixed device/circuit simulation of an SEU, the struck device is modeled in the “device domain” using multi-dimensional device simulation, whereas the rest of the circuit is represented by SPICE-like compact circuit models. The two domains are tied together by the boundary conditions at contacts, and the solution to both sets of equations is rolled into one matrix solution [19,21]. The advantage is that only the struck device is modeled in multiple dimensions, whereas the rest of the circuit...
consists of computationally efficient SPICE models. This decreases simulation times and greatly increases the complexity of the external circuitry that can be modeled.

However, as circuits grow exponentially in density and complexity, comprehensive mixed device/circuit simulation is impractical. Therefore in our approach, we stick to circuit level simulation with analytic current models to perform a systematic and exhaustive laser scanning examination.

3. SIMULATION METHODOLOGY

The traditional procedure to design and test an IC against laser fault injection attacks is illustrated in Figure 4(a). A major concern is that security evaluation occurs too late in the design cycle to allow for efficient repair. If deficiencies are identified during the test phase, costly design re-spins are demanded. As a comparison, the procedure with evaluation incorporated in the design flow is demonstrated in Figure 4. This design flow can spot design oversights or errors at an early stage to avoid costly silicon re-spins.

3.1. n-Type versus p-type complementary metal-oxide-semiconductors

As shown by Dodd et al. [22], with increasing LET of incident ion, the most sensitive area is found to be the reverse-biased diffusion of n-MOS. Then, for higher LET, the reverse-biased diffusion of the p-MOS becomes sensitive as well. The optical attack is substantially more effective at turning on n-type transistors than their p-type counterparts, the laser radiation will result in one of three behaviors in a given logic gate:

- The laser radiation is not strong enough to cause either the n-type or the p-type CMOS transistors to conduct, so no state change occurs at the logic cell output.
- The laser radiation switches on the n-type but not the p-type transistors, so abnormal behavior may occur.
- The laser radiation is strong enough to cause both n-type and p-type CMOS transistors to conduct in a logic gate. This results in a strong leakage current or even a strong VDD-to-GND short circuit, which may damage the circuit eventually if no current limit protection is provided.

Of the three behaviors, only the second is considered as a successful attack as opposed to sabotage and is therefore the focus of this simulation methodology. This allowed us to simply focus on n-type transistors in the simulation of security evaluation targeting Type I attackers. Apparently, in the case where the laser can target a single p-type transistor and successfully switch it on, the attacker is able to manipulate the circuit more capably. This situation falls into the category of Type II and III attacks. The corresponding simulation requires layout scans over every single transistor.

3.2. Simulation procedure

The simulation procedure against laser fault injection attacks is illustrated in Figure 5. Co-simulation of circuit level and transistor level is conducted through combining a Verilog simulator (or simulators supporting other hardware description languages (HDLs)) and a SPICE-like simulator. The modules of interest in the Verilog netlist are swapped out with the full transistor-level netlist. Within the transistor-level netlist, the cells under attack are instantiated into transient stimuli according to the layout scanning process. The stimuli are in essence voltage pulses supplied to state buffers to the nodes under attack. The HDL/SPICE integration allows the simulation to have gate-level speed and transistor-level accuracy. The scanning process in this paper is performed with Silicon Ensemble™ (Cadence Design Systems, San Jose, CA, USA), and the HDL/SPICE co-simulator is chosen to be NANOsim™ (Synopsys Inc., Mountain View, CA, USA) integrated with the Synopsys Verilog simulator, VCS™ (Synopsys Inc., Mountain View, CA, USA). Other similar and commercially available simulation environments include AMS™ (Cadence Design Systems, San Jose, CA, USA), ADVANCE™ (Mentor Graphics, Wilsonville, OR, USA), SMASH™ (Dolphin Integration, Genoble, France), and so on.

The layout can be scanned with any size of laser illumination spot, which can target from a single transistor to hundreds of transistors, depending on the equipment used by the attackers as described in Section 2. The scans can be performed over a particular area such as the ALU, register file, or even the whole processor. Figure 6 illustrates a scanning in simulation, where each scan (S11, S12 ..., Smax) generates a list of logic cells under attack. For example, in a particular scan, exposed cells are listed as follows:

```
m/datapath/U355  m/datapath/fi.reg.4  m/U1490  m/U1506
m/datapath/alu/U33  m/U1458  FC.299  m/U1223
```

Also, the optical attack is more effective at turning on p-type transistors than n-type, depending on the process technologies, especially the substrate type and the well type, see Section 2.2 for details.
Among the selected cells, FC_299 is a filler cell, and the rest are logic cell instances. We first discard the filler cells, then check the standard cell library, mapping the logic cells to their internal nodes, especially the nodes connected to n-type transistors, or more effective at turning on p-type transistors than n-type, depending on the process technologies, especially the substrate type and the well type [23]. In addition to what may be considered a useful attack mechanism, negative effects are also possible. These include the possibility that latch-up may be induced by the generation of photocurrents in the bulk (the substrate and the well). Of less concern when using readily available infrared and visible laser light sources is the ionization of gate and field oxides due to the large band gap energy of silicon dioxide (which would require a laser with a wavelength in the UV-C range). Ionization of this type is common when higher energy forms of radiation are absorbed. The subsequent accumulation of positive charge results in a long term shift in transistor characteristics.

After obtaining the list of exposed cells for each scan, we then supply the internal nodes with transient voltage pulses via tri-state buffers. The enable signal of the tri-state buffer is synchronized with the target instruction execution during a program operation. The Xin signal is the transient voltage pulse, whose time profile is the same as the laser illumination-induced current profile, which can be double-exponential, analytical, or even piecewise linear pulse, depending on the accuracy demand of the circuit simulation. The Yout signal connects to internal nodes of the exposed cells. Figure 7 demonstrates the connection of a tri-state buffer to the internal nodes of the exposed cell m/U1490. The fragment of a SPICE simulation program, shown in Figure 11, is used for the connection of the Yout signal to the internal nodes of the exposed cells out of a laser scanning (Figure 8).

The co-simulation integrates the voltage pulses and illuminated cells in SPICE, whereas the rest of the circuit remains in HDLs. Analyzing the response and comparing it to that of the normal operation, we can evaluate the security of the circuit design against laser fault injection attacks. If it fails, modification or even redesign of the circuit is required. If it passes, then designers can continue to have the chip manufactured.

4. RESULTS

4.1. Laser scanning image on pn junctions

A passive laser scanning operation was applied to the sample. In failure analysis, this technique is called optical beam-induced current (OBIC). OBIC imaging is very sensitive to
electron-hole recombination. The result of scanning the pn junctions of the sample with the laser is presented in Figure 9. Figure 9(a) demonstrates the surface image of the four pn junctions, and Figure 9(b) demonstrates the OBIC image of the pn junctions. The bright area in Figure 9(b) indicates two of the four pn junctions are conducted.

4.2. Optical attack simulation results

Simulation of optical fault injection attacks has been carried out on the Springbank test chip. This simulation addresses the synchronous processor (S-XAP) on the top left corner of the chip as shown in Figure 10. The substrate/well formation is a p-substrate with twin-well. According to Section 2.2, n-type transistors are easier to switch on so are simulated.

The aim of the test is to exhaustively examine the ALU and decoder of processor S-XAP to determine if it is susceptible to optical fault injection attacks. We target simple instructions (e.g. XOR, shift, load, store), which can give a good indication of how the hardware reacts to operations of application programs. A fragment of a program, shown in Figure 11, is used for the evaluation, where the processor loads the first argument to register AH, XOR it with the second argument from memory, then saves the result back to memory. The laser attack is synchronized with the XOR operation, meaning the transient voltage sources will be activated at this moment in simulation.

The simulation procedure is implemented as introduced in Section 3. The ALU and the decoder are scanned with a scanning square size of about 300 µm², to cover 10–15 logic cells. Figure 12 shows the screenshot of the scanning procedure. The spot is moved within the area horizontally each time by one cell width (about 4 µm or more), then vertically by one cell height (about 6 µm). The scanning produces 120 lists of cells, mimicking 120 optical fault injection attacks. For each list, we connect the internal nodes to transient voltage sources and incorporate the stimuli into the SPICE netlist. Then, the Verilog/HSPICE co-simulation running the above simple instruction program is performed to examine the circuit response during each optical attack.

The exhaustive examination of the 120 simulation runs shows different results:

1. The processor results in a deadlock in many cases, which is desirable in terms of security, provided this does not leak secret information.

Figure 8. Fragment of the VCS simulation program to supply the internal nodes of the exposed cells with transient voltage pulses.

Figure 9. Laser scanning image on pn junctions of the sample: (a) surface image; (b) conducted image.
ld ah,@(1,x) ; load first argument
nop
nop
xor ah,@(2,x) ; XOR operation
nop
nop
st ah,@(3,x) ; save result

Figure 11. Fragment of the instruction program used for the evaluation.

Figure 10. The microprocessor (S-XAP) on the chip is under simulation test.

Figure 12. Screen shot of scanning procedure over the layout of ALU and decoder of processor S-XAP: the region within the smaller square being illuminated.
(2) Some other cases show normal program execution. This implies the introduced fault may be part of the “don’t care” state of the subsequent operation of the circuit [6].

(3) Two failures are also revealed:
(a) The first disrupts the XOR operation by changing the value in the AH register.
(b) The second failure causes a memory dump. Instead of executing a data write-to-memory, the processor keeps reading the contents of the whole memory. We suspect the memory dump occurs when the decoder was struck in the test. One bit of the decoder output is set high, which indicates the consecutive LOAD operation in ALU.

Modifying register values implies that setting part of the key to a known value becomes feasible to the attackers. In most digital signature algorithms, with multiple signatures and partial knowledge of the private key, the whole key could be recovered [26]. Dumping memory can be dangerous to designs implemented with an architecture where a single storage structure is used to hold both instructions and data. If the memory contains passwords and decryption keys, then by carefully analyzing the dumped memory, one can break the cryptographic device. In contrast, a design implemented with Harvard architecture [25] could offer better protection against microprobing attacks, as it uses physically separate storage for instructions and data. The same trick applied to a Harvard microcontroller would reveal only the program code, whereas the data memory containing sensitive information will not be available.

It takes about 10 min to run the scanning process (containing 120 scans) with Cadence Silicon Ensemble™. Then, it takes about 4 h to complete the 120 runs of HDL/SPICE co-simulation, with each run to have 14,000 transistors simulated in Synopsys NanoSim™ and the rest 10s of logic gates simulated in Synopsys VCS™. The scan time can be limited by the value of the total scan area divided by the size of the laser illumination spot. The effective scan time can be further reduced by selecting the most critical and vulnerable area. In fact, how to select the most critical area to laser fault injection attacks is one of the intentions of the design-time evaluation methodology proposed in this paper. All the simulation work is done on a 1.6-GHz AMD Athlon processor (Advanced Micro Devices, Sunnyvale, CA, USA) with a 2-GB memory.

4.3. Experimental results

A laser fault injection experiment was conducted by Gemplus® on the same test chip to provide a side-by-side comparison [24]. The test chip was fabricated in the UMC 0.18 μm six metal CMOS process. The test chip was mounted in a zero-insertion force socket, which was mounted on the bottom side of the test board, thus easing access for the laser attack. The laser is synchronized with the executed program (same as the code used in the simulation, except that in the experiment, the assembly code contained a subroutine to display the two operands and the result of the XOR operation) via an interrupt signal from a particular I/O pin. The experiments reveal that not every portion of the processor is sensitive to the laser. When there is an actual effect, the processor goes into a failure state in most cases, and the chip has to be reset in order to reload the program (Figure 13).

By shooting the laser at the ALU of the processor, we finally obtained effects like modification of the result of an XOR operation, which agrees with the first type of failure discovered through simulation. Also, we succeeded in dumping the data memory in the processor S-XAP by shooting the laser at a place within the region of the ALU and the registers. This result is similar to the simulation. The attack is based on the fact that n-MOS is more easily turned on than p-MOS, due to low-threshold LET. The larger the difference between p-MOS SET and n-MOS LET, the easier for the laser fault injection.

The disrupted execution had the effect of outputting consecutive values from data memory. This is because the laser fault injection causes temporary logic failures. When such a logic failure recovers, the “normal” flow of the program is resumed, but in the meantime, the data stored in control registers seemed to have changed, which results a large part of the RAM being dumped onto the serial port of the chip. In our test chip, the value contained in the register containing the size of the UART’s buffer had been corrupted leading to larger part of the RAM being dumped. This feature is quite common in design for debugging. We discovered that it causes the chip to be vulnerable in laser fault injection attacks instead.

![Figure 13. Laser experiment on ALU and decoder of processor S-XAP.](image-url)
5. CONCLUSION

A simulation methodology has been proposed to evaluate the dependability of ICs against laser fault injection attacks at design time. This simulation methodology involves exhaustive scans over the layout with variable laser spot size according to the attack scenario. Logic gates under illumination are identified and simulated in SPICE with additional voltage spikes at appropriate nodes that mimic the attack. This SPICE model is co-simulated with the rest of the system represented in Verilog.

Simulation performed on our test chip has demonstrated that the laser fault injection could modify the value stored in registers so that setting part of the key to a known value becomes feasible to the attackers. Attacks on other areas also caused a data memory dump, which can be extremely dangerous if the memory contains passwords or secret keys. Experimental results revealed the same kind of weaknesses, which gives us the confidence in the proposed simulation methodology.

ACKNOWLEDGEMENTS

This work was supported by the National Natural Science Foundation of China (grant no. 60901052) and the Fundamental Research Project of Shenzhen (grant no. JC200903160412A). Instruction and help from Dr. Simon Moore at University of Cambridge are highly appreciated.

REFERENCES


Dear Author,

During the copyediting of your paper, the following queries arose. Please respond to these by annotating your proofs with the necessary changes/additions.

- If you intend to annotate your proof electronically, please refer to the E-annotation guidelines.
- If you intend to annotate your proof by means of hard-copy mark-up, please refer to the proof mark-up symbols guidelines. If manually writing corrections on your proof and returning it by fax, do not write too close to the edge of the paper. Please remember that illegible mark-ups may delay publication.

Whether you opt for hard-copy or electronic annotation of your proofs, we recommend that you provide additional clarification of answers to queries by entering your answers on the query sheet, in addition to the text mark-up.

<table>
<thead>
<tr>
<th>Query No.</th>
<th>Query</th>
<th>Remark</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q1</td>
<td>AUTHOR: Please provide a suitable figure (abstract diagram or illustration selected from the manuscript or an additional eye-catching figure) and a short “GTOC” abstract (maximum 80 words or 3 sentences) summarizing the key findings presented in the paper for Table of Content (TOC) entry.</td>
<td><img src="https://example.com/remark.png" alt="Remark" /></td>
</tr>
<tr>
<td>Q2</td>
<td>AUTHOR: “Very-large-scale integration”. Is this the correct definition for VLSI? Please change if this is incorrect.</td>
<td><img src="https://example.com/remark.png" alt="Remark" /></td>
</tr>
<tr>
<td>Q3</td>
<td>AUTHOR: “Temperature variation using an alcoholic cooler until the chips temperature to a value where write operations work but reads dont or the other way around, a number of attacks can be mounted.” This sentence has been reworded for clarity. Please check and confirm it is correct.</td>
<td><img src="https://example.com/remark.png" alt="Remark" /></td>
</tr>
<tr>
<td>Q4</td>
<td>AUTHOR: Section 5 presents a discussion of defense technologies followed by a brief conclusion in Section 6. In this sentence, a discussion section was mentioned but is not present in the paper. The conclusion section has been captured as Section 6, and please advise as to what section should be captured as Discussion section.</td>
<td><img src="https://example.com/remark.png" alt="Remark" /></td>
</tr>
<tr>
<td>Q5</td>
<td>AUTHOR: “gallium arsenide”. Is this the correct definition for GaAs? Please change if this is incorrect.</td>
<td><img src="https://example.com/remark.png" alt="Remark" /></td>
</tr>
<tr>
<td>Q6</td>
<td>AUTHOR: “Complementary metal–oxide–semiconductor”. Is this the correct definition for CMOS? Please change if this is incorrect.</td>
<td><img src="https://example.com/remark.png" alt="Remark" /></td>
</tr>
<tr>
<td>Q7</td>
<td>AUTHOR: Please give manufacturer information for SPICE: company name, town, state (if USA), and country.</td>
<td><img src="https://example.com/remark.png" alt="Remark" /></td>
</tr>
<tr>
<td>Q8</td>
<td>AUTHOR: Developer information has been provided for Silicon Ensemble. Please change if this is incorrect.</td>
<td><img src="https://example.com/remark.png" alt="Remark" /></td>
</tr>
<tr>
<td>Query No.</td>
<td>Query</td>
<td>Remark</td>
</tr>
<tr>
<td>----------</td>
<td>-------</td>
<td>--------</td>
</tr>
<tr>
<td>Q9</td>
<td>AUTHOR: Please give manufacturer information for NanoSim: company name, town, state (if USA), and country.</td>
<td></td>
</tr>
<tr>
<td>Q10</td>
<td>AUTHOR: Developer information has been provided for VCS. Please change if this is incorrect.</td>
<td></td>
</tr>
<tr>
<td>Q11</td>
<td>AUTHOR: Developer info has been provided for AMS. Please change if this is incorrect.</td>
<td></td>
</tr>
<tr>
<td>Q12</td>
<td>AUTHOR: Developer information has been provided for ADVANCE MS. Please change if this is incorrect.</td>
<td></td>
</tr>
<tr>
<td>Q13</td>
<td>AUTHOR: Developer information has been provided for SMASH. Please change if this is incorrect.</td>
<td></td>
</tr>
<tr>
<td>Q14</td>
<td>AUTHOR: Please define ALU.</td>
<td></td>
</tr>
<tr>
<td>Q15</td>
<td>AUTHOR: Figure 8 was not cited in the text. An attempt has been made to insert the figure into a relevant point in the text – please check that this is OK. If not, please provide clear guidance on where it should be cited in the text.</td>
<td></td>
</tr>
<tr>
<td>Q16</td>
<td>AUTHOR: Please define AH.</td>
<td></td>
</tr>
<tr>
<td>Q17</td>
<td>AUTHOR: Manufacturer information has been provided for 1.6 GHz AMD Athlon processor. Please change if this is incorrect.</td>
<td></td>
</tr>
<tr>
<td>Q18</td>
<td>AUTHOR: Please define UMC.</td>
<td></td>
</tr>
<tr>
<td>Q19</td>
<td>AUTHOR: Figure 13 was not cited in the text. An attempt has been made to insert the figure into a relevant point in the text – please check that this is OK. If not, please provide clear guidance on where it should be cited in the text.</td>
<td></td>
</tr>
<tr>
<td>Q20</td>
<td>AUTHOR: Please define UART.</td>
<td></td>
</tr>
<tr>
<td>Q21</td>
<td>AUTHOR: References [11]; [18]; and [20] was not been cited in the text. Please indicate where it should be cited; or delete from the Reference List and renumber the References in the text and Reference List.</td>
<td></td>
</tr>
<tr>
<td>Q22</td>
<td>AUTHOR: Please provide authorship for reference 11.</td>
<td></td>
</tr>
<tr>
<td>Q23</td>
<td>AUTHOR: Please provide publishing year and date of access for reference 25.</td>
<td></td>
</tr>
</tbody>
</table>
USING E-ANNOTATION TOOLS FOR ELECTRONIC PROOF CORRECTION

Required Software
Adobe Acrobat Professional or Acrobat Reader (version 7.0 or above) is required to e-annotate PDFs. Acrobat 8 Reader is a free download: http://www.adobe.com/products/acrobat/readstep2.html

Once you have Acrobat Reader 8 on your PC and open the proof, you will see the Commenting Toolbar (if it does not appear automatically go to Tools>Commenting>Commenting Toolbar). The Commenting Toolbar looks like this:

![Commenting Toolbar](image)

If you experience problems annotating files in Adobe Acrobat Reader 9 then you may need to change a preference setting in order to edit.

In the “Documents” category under “Edit – Preferences”, please select the category ‘Documents’ and change the setting “PDF/A mode:” to “Never”.

**Note Tool — For making notes at specific points in the text**
Marks a point on the paper where a note or question needs to be addressed.

**Replacement text tool — For deleting one word/section of text and replacing it**
Strikes red line through text and opens up a replacement text box.

**Cross out text tool — For deleting text when there is nothing to replace selection**
Strikes through text in a red line.
Approved tool — For approving a proof and that no corrections at all are required.

How to use it:
1. Click on the Stamp Tool in the toolbar
2. Select the Approved rubber stamp from the ‘standard business’ selection
3. Click on the text where you want to rubber stamp to appear (usually first page)

Highlight tool — For highlighting selection that should be changed to bold or italic.

How to use it:
1. Select Highlighter Tool from the commenting toolbar
2. Highlight the desired text
3. Add a note detailing the required change

Attach File Tool — For inserting large amounts of text or replacement figures as a files.

How to use it:
1. Click on paperclip icon in the commenting toolbar
2. Click where you want to insert the attachment
3. Select the saved file from your PC/network
4. Select appearance of icon (paperclip, graph, attachment or tag) and close

Pencil tool — For circling parts of figures or making freeform marks

How to use it:
1. Select Tools > Drawing Markups > Pencil Tool
2. Draw with the cursor
3. Multiple pieces of pencil annotation can be grouped together
4. Once finished, move the cursor over the shape until an arrowhead appears and right click
5. Select Open Pop-Up Note and type in a details of required change
6. Click the X in the top right hand corner of the note box to close.
Help

For further information on how to annotate proofs click on the Help button to activate a list of instructions: