CHAPTER

PHYSICAL ATTACKS AND COUNTERMEASURES

10

CONTENTS

Hardware Security. https://doi.org/10.1016/B978-0-12-812477-2.00015-0

10.1 INTRODUCTION

Physical attacks are divided into three categories: noninvasive, semi-invasive, and invasive attacks. A noninvasive attack does not require any initial preparations of the device under test, and will not physically harm the device during the attack. The attacker can either tap the wires to the device, or plug it into a test circuit for the analysis. Invasive attacks require direct access to the internal components of the device, which normally requires a well-equipped and knowledgeable attacker to succeed. Meanwhile, invasive attacks are becoming constantly more demanding and expensive, as feature sizes shrink, and device complexity increases. There is a large gap between noninvasive and invasive attacks. Many attacks fall into this gap, called semi-invasive attacks. They are not very expensive as classical penetrative invasive attacks, but are as easily repeatable as noninvasive attacks. Like invasive attacks, they require depackaging the chip in order to get access to its surface. However, the passivation layer of the chip remains intact, as semi-invasive methods do not require creating contacts to the internal wires. This chapter mainly focuses on invasive physical attacks. Reverse engineering, microprobing attack, and invasive fault injection attack are the most common physical attacks, and will be introduced, respectively, in the rest of this chapter.

10.2 REVERSE ENGINEERING

Reverse engineering (RE) is the process involving the thorough examination of an object to achieve a full understanding of its construction and/or functionality; a method used by attackers as part of mounting their attack. RE is now widely used to clone, duplicate, or reproduce systems and devices in various security-critical applications, such as smartcards, smartphone, military, financial, and medical systems [1]. In this section, the RE of electronic systems, which can be achieved by extracting the system's underlying physical information through destructive or nondestructive methods, is discussed [2,3].

The motivation for RE could be "honest" or "dishonest," as shown in Table 10.1 [4–6]. Those with honest motivations tend to perform RE for verification, fault analysis, research, and education of an existing product. In many countries, RE is legal, as long as patents and design copyrights are not violated [7]. However, RE could be used to clone, pirate, or counterfeit a design, to develop an attack, or to insert a hardware Trojan. Such actions are considered dishonest. If the functionality of a cloned system is close enough to the original one, then the dishonest entity, or individuals, could sell large amounts of counterfeit products without prohibitive research and development costs required by the IP owner [8]. One example of dishonest RE took place during World War II. An American B-29 bomber was captured, reverse engineered, and cloned by the former Soviet Union (Tupolev Tu-4 bomber) [9]. The original and the cloned bombers are shown in Fig. 10.1. The configuration of the two bombers are almost the same, except for the engines and cannons.

Aside from RE of large systems, sensitive data, such as critical design parameters and personal secret information can also be extracted, or cloned, from electronic chips and printed circuit boards (PCBs). For example, it is quite easy to reverse engineer a PCB, because of its simple architecture and

Table 10.1 Motivation for reverse engineering

"Honest" motivations	"Dishonest" motivations
Failure analysis and defect identification	Fault injection attacks
Detection of counterfeit products [5,8]	Counterfeiting
Circuit analysis to recover manufacturing defects	Tampering
Confirmation of IP	IP piracy and theft
Hardware Trojan detection [6]	Hardware Trojan insertion
Analysis of a competitor's/obsolete product	Illegal cloning of a product
Education and research	Development of attacks

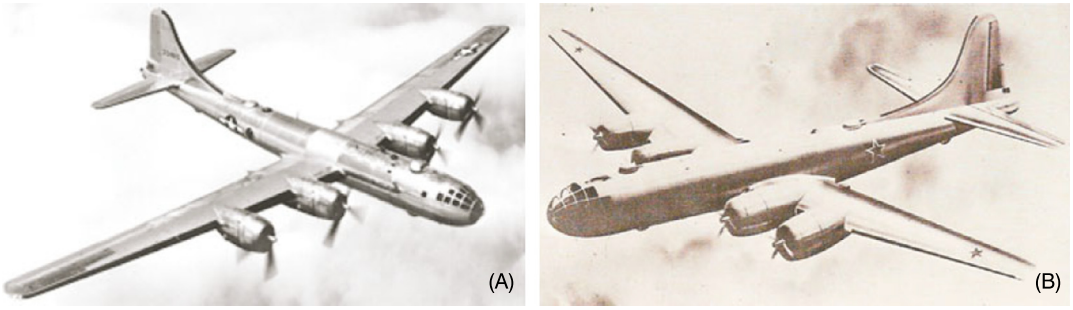

FIGURE 10.1

An example of RE from World War II: (A) a U.S. Air Force B-29 bomber, and (B) a Soviet Union Tupolev Tu-4 bomber, a reverse-engineered copy of the B-29.

increasing reliance on commercial off-the-shelf components. RE of PCBs and ICs could also provide opportunities for the future attacks against them. For example, many smartcards today contain ICs that store personal information and perform transactions. Dishonest parties could reverse engineer these ICs to access the confidential information of the card holder, commit financial crimes, and so forth.

Another concern in electronics industry is IC piracy through RE [10]. In 2010, Semiconductor Equipment and Materials International (SEMI) published a survey about IP infringement. The survey revealed that 90% of semiconductor companies experienced IP infringement, and 54% of them faced serious infringement on their products [11]. Many dishonest companies can illegally clone the circuits and techniques to mass produce, and sell those pirated copies in open market without authorization. The latter results in unrecoverable losses to the IP owner. Counterfeit ICs and systems may also be tampered, leading to vulnerabilities and life-threatening issues.

To summarize, RE is a long-standing problem that is of great concern to today's governments, militaries, various industries, and individuals due to: (1) the attacks and security breaches that could occur through the RE of classified systems, such as those of the military and financial institutions; (2) the safety issues and costs resulting from unintended use of counterfeit products in critical systems and infrastructures; (3) the loss in profits and reputation for IP owners; and (4) the negative impact that RE has on new product innovations, and incentives for research and development.

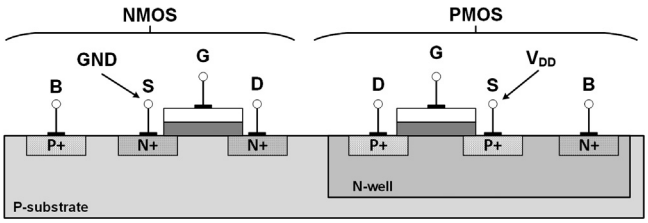

FIGURE 10.2

Simplified cross-sectional view of CMOS transistors.

As a result of these concerns, researchers, companies, and the defense departments of many nations are persistently seeking anti-RE techniques to prevent adversaries from accessing their protected products and systems. For example, the U.S. DoD is currently conducting research on anti-RE technologies that may prevent classified data, weapons, and IP from being compromised by foreign adversaries [12]. The objective of the DoD's antitamper program is to obstruct unapproved technology transfer, maximize the costs of RE, enhance U.S.'s coalition military capacities, train the DoD community, and educate the DoD community on antitampering technologies [13]. Unfortunately, most of this task is classified, and therefore is not available to the industrial sector or the wider research community.

Anti-RE techniques should have the ability to monitor, detect, resist, and react to invasive and noninvasive attacks. Several techniques could be used as anti-RE. For example, tamper-resistant materials and sensors have been used to resist theft or RE [14]. Hard barriers like ceramics, steel, and bricks have been used to separate the top layer of the electronic devices, so that tampering or RE attempts might be foiled by the destruction of the protective devices. To protect against microprobing attempts, single chip coatings have also been applied. Many different packaging techniques could also be used to protect a device: brittle packages, aluminum packages, polished packages, and bleeding paint, and holographic and other tamper-responding tapes, and labels [14]. Sensors of interest include voltage sensors, probe sensors, wire sensors, PCB sensors, motion sensors, radiation sensors, and top-layer sensor meshes. Materials like epoxy with potting, coating, and insulation have been used to block x-ray imaging attempts.

In addition, obfuscation software and hardware security primitives have been used for the protection of systems and software. These anti-RE techniques can be helpful for protecting confidential information from different types of RE attempts. Some other methods for protecting these systems are as follows: bus encryption, secure key storage, side-channel attack (SCA) protection, and tamper-responding technology [14,15].

Following is a presentation on the RE of electronic devices from chip to system levels:

(1) Chip-level RE: A chip is an IC comprised of electronic devices that are fabricated using semiconductor material. A chip has package material, bond wires, a lead frame, and die. Each die has several metal layers, vias, interconnections, passivation, and active layers [16]. In Fig. 10.2, a simplified cross-sectional view of NMOS and PMOS is shown respectively. As shown in this figure, polysilicon gates (G) of NMOS and PMOS transistors are connected together somewhere off the page to form the input of the inverter. The source (S) of the PMOS of the inverter is connected to a metal Vdd line, and the source of the NMOS is connected to a metal ground (GND) line. The drains (D) of the PMOS

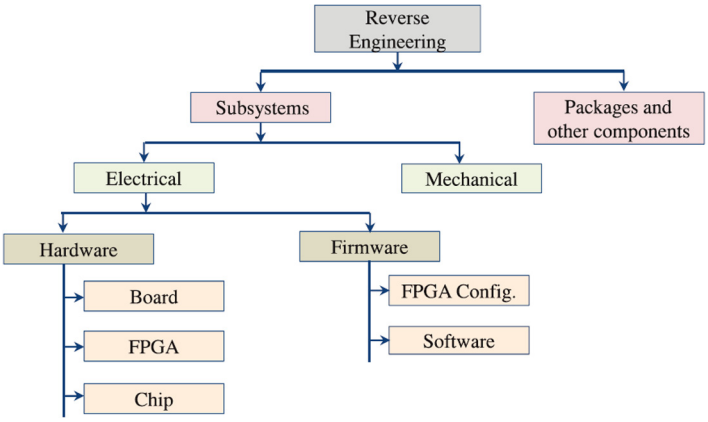

FIGURE 10.3

Taxonomy of RE.

and NMOS are connected together with a metal line for the output of the CMOS inverter. The chip could be analog, digital, or mixed signal. Digital chips include application-specific integrated circuits (ASICs), field-programmable gate arrays (FPGAs), and memories. RE of chips can be nondestructive or destructive. X-ray tomography is a nondestructive method of RE that can provide layer-by-layer images of chips, and is often used for the analysis of internal vias, traces, wire bonding, capacitors, contacts, or resistors. Destructive analysis, on the other hand, might consist of etching and grinding every layer for analysis. During the delayering process, pictures are taken by either a scanning electron microscope (SEM), or a transmission electron microscope (TEM).

(2) PCB-level RE: Electronic chips and components are mounted on a laminated nonconductive PCB [17] and electrically interconnected using conductive copper traces and vias. The board might be single or multilayered, depending on the complexity of the electronic system. RE of PCBs begins with the identification of the components mounted on the board, its traces on the top and bottom (visible) layers, its ports, and so forth. After that, delayering or x-ray imaging could be used to identify the connections, traces, and vias of the internal PCB layers.

(3) System-level RE: Electronic systems are comprised of chips, PCBs, and firmware. A system's firmware includes the information about the system's operation and timing, and is typically embedded within nonvolatile memories (NVMs), such as ROM, EEPROM, and Flash. For more advanced designs with FPGAs (for example, Xilinx FPGAs), the firmware-like netlists are also stored within the NVM memories (also called bitstream). By reading out and analyzing the contents in the memory, RE can provide a deeper insight into the system under attack.

Based on the preceding discussions, a comprehensive taxonomy of RE is shown in Fig. 10.3. First, RE is performed to tear down the product or system to identify the subsystems, packages, and other components. The subsystems could be electrical or mechanical. In this chapter, only electrical subsystems are focused. The electrical subsystems under analysis consist of hardware and firmware. A reverse engineer could analyze the FPGA, board, chip, memory, and software to extract all information. This effort is concerned with RE when it is done with malicious intentions, and with anti-RE as a remedy

against this form of RE. This type of RE and anti-RE for each level, including equipment, techniques, and materials, is examined.

10.2.1 EQUIPMENT

Advanced RE requires a set of specialized equipment. Below, a short summary of some of the equipment commonly used in RE is presented as shown in Figs. 10.4, 10.5, and 10.6.

Optical high/super-resolution microscopy (digital). The limitations of conventional digital microscopy include limited depth of field, a very thin focus field, and keeping all parts on an object simultaneously in focus [18]. To overcome these limitations, optical high-resolution microscopes are now being used. Optical super-resolution microscopes take a series of images and put them together to create a 3D image that reflects different heights. However, optical microscopes can only be used to analyze PCB and chip exteriors, as the resolution is too low for current chip feature sizes ($\ll$ 100 nm).

Scanning electron microscopes. In a SEM, focused beams of electrons are used to produce images [22]. For a sample, the electrons interact with atoms, a process that produces signals for detection. It is expected that reverse engineers would start with a cross section of an unknown chip. SEM could be used for analyzing the cross section, and the composition and thickness of each layer of the die. The object could be magnified by 10 times, to approximately 30,000 times. SEM provides the following advantages over traditional microscopes:

– *Higher resolution*: SEM has higher resolution, and with high magnification, it can resolve the features on the submicron level.

– *Large depth of field*: When a specimen (for example, the internal elements of a chip) is focused on for an image, the height of the specimen is called the depth of field. The SEM has a depth of field that is more than 300 times greater than that of a light microscope, which means that a specimen's otherwise unobtainable details can be obtained with a SEM.

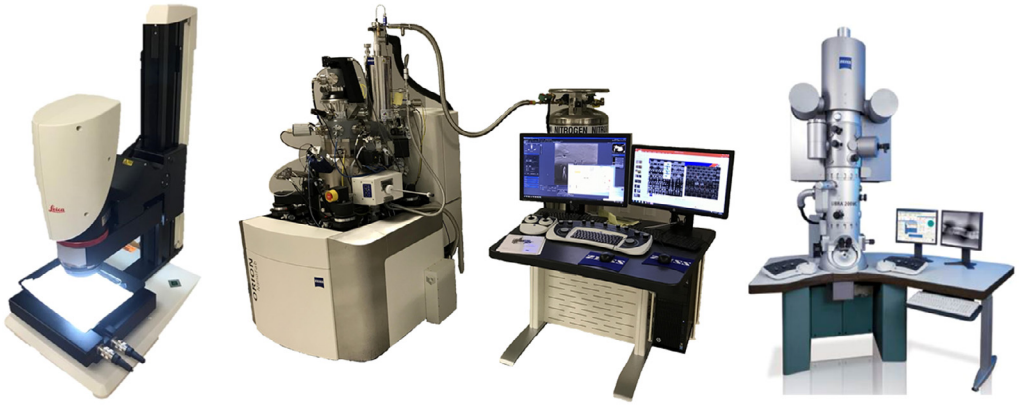

FIGURE 10.4

(A) Optical microscopy. (B) Scanning electron microscope (SEM). (C) Transmission electron microscope (TEM).

(A) (B) (C)

FIGURE 10.5

(A) Focused ion beam (FIB). (B) High-resolution x-ray microscopy. (C) Probe station.

(A) (B)

FIGURE 10.6

(A) Logic analyzer. (B) Computer numerical control (CNC) [30].

Transmission electron microscopes. With TEMs, a beam of electrons is transmitted through and interacts with a sample [21,23]. Like SEMs, TEMs have a very high spatial resolution, which can provide detailed information about the internal structures of a sample [24]. In addition, a TEM can be used to view a chip's cross-section and its internal layers.

Focused ion beam. The working principle of a focused ion beam (FIB) is the same as a SEM, but instead of using an electron beam, an ion beam is used. The ion beam enables one to perform material deposition and removal with nanometer resolution, which can be used for TEM sample preparation, and circuit editing. There are different types of ion sources for the ion beam, but the most popular one is gallium (Ga) liquid metal. The new generation of these tools is called *plasma* FIB (PFIB), which works at a higher power, and results in shorter material processing time.

Scanning capacitance microscopy. For the illustration of dopant profiles on the 10 nm scale of semiconductor devices, scanning capacitance microscopy (SCM) is used because of its high spatial resolution [3]. A probe electrode is applied at the top of the sample surface, and this electrode then

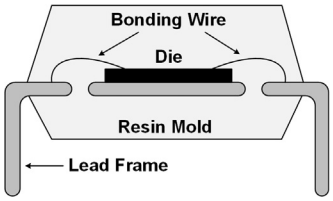

FIGURE 10.7

Cross-sectional view of IC parts.

scans across the sample. The change in electrostatic capacitance between the surface and the probe is used for obtaining information about the sample [28].

High-resolution x-ray microscopy. X-ray microscopy is used to nondestructively test a sample, such as a chip or a PCB board. With this method, x-rays are used to produce a radiograph of the sample, which shows its thickness, assembly details, holes, vias, connectors, traces, and any defects that might be present [28].

Probe stations. Probe station supports a wide variety of electrical measuring, device and wafer characterization, failure analysis, submicron probing, optoelectronic engineering tests, and more. There are up to 16 positioners in these kinds of systems located on a vibration isolated frame, which stabilizes the platen. These features enable a highly reliable and repeatable testing process down to the submicron level. A pull-out vacuum chuck stage holds the testing samples and the motorized platen, whereas the chuck and positioners provide enough flexibility to perform tests on many different samples.

Logic analyzers. A logic analyzer is an electronic instrument that can observe and record multiple signals on a digital system or digital circuits simultaneously. The use of a logic analyzer can facilitate RE at the chip, board, and system levels. In the case of FPGA bitstream RE, the logic analyzer can be adopted to measure the JTAG communication signals between the FPGA and external memory.

Computer numerical control. The need for automating machining tools, which are typically controlled manually, led to the creation of the computer numerical control (CNC), where computers control the process [30]. CNCs can run mills, lathes, grinds, plasma cutters, laser cuts, and so forth. The motion is controlled along all three main axes, which enables a 3D process.

10.2.2 CHIP-LEVEL RE

An IC typically consists of a die, a lead frame, wire bonding, and molding encapsulant, as shown in Fig. 10.7. The package of a chip can be classified in different ways. The materials that are used can be ceramic or plastic [31]. Considering that ceramics are costly, plastics are commonly used as the package material. Packaging can also be wire bond or flip chip [32]. In wire bond packaging, wires are connected to the lead frame. There are several types of wire bonding: concentric bond rings, double bonds, and ball bonding. In contrast, flip-chip packaging is a technique that allows for a direct electrical connection between face-down ("flipped" so that its top side faces down) electronic components, and substrates, circuit boards, or carriers. This electrical connection is formed from conductive solder bumps instead of wires. Flip chips have several advantages over wire bond packaging: superior

electrical and thermal performance, higher input-output capability, and substrate flexibility. However, flip-chips are often considered more costly than wire bonds [32].

At the chip level, the goal of the RE process is to find package materials, wire bonding, different metal layers, contacts, vias and active layers, and interconnections between metal layers. The RE process has several different steps:

– *Decapsulation:* Decapsulation exposes the internal components of the chip, which allows for the inspection of the die, interconnections, and other features.

– *Delayering:* The die is analyzed layer by layer, destructively, to see each metal, passivation, poly, and active layer.

– *Imaging:* An image is taken of each layer in the delayering process by using SEM, TEM, or SCM.

– *Post-processing:* In this process, the images from the previous step are analyzed, schematic and high-level netlists are created for functional analyses, and the chip is identified. Each of these steps is discussed in greater detail in the following sections.

10.2.2.1 Decapsulation

First, reverse engineers identify the package materials and remove the chip's packaging. Depot is the traditional method by which an acid solution is used for removing the package [3]. A package may be made from different kinds of materials, so one has to be precise when choosing the acid. These acid solutions are used to etch off the packaging material without damaging the die and interconnections. Mechanical and thermal methods are used to remove a die from ceramic packages. These methods are applied to both polish the ceramic materials and remove the lids [3].

To remove the die package, one can use selective or nonselective methods. Wet chemical etching and plasma etching can be used as selective techniques, whereas nonselective techniques would be thermal shock, grinding, cutting, and laser ablation. Different kinds of decapsulation methods and their pros and cons are shown in Table 10.2.

After decapsulation, the die needs to be cleaned before delayering and/or imaging can be performed, because dust may be present, resulting in artifacts [33]. Different methods for cleaning the dust are outlined next:

– *Spray cleaning:* A syringe filled with acetone is attached to a very fine blunt-tip needle. The syringe is then used to spray particles off of the die.

– *Acid cleaning:* To remove organic residues, fresh acid can be used after decapsulation.

– *Ultrasonic cleaning:* Water, detergent (lab grade), or solvents can be used for ultrasonic cleaning after bare die decapsulation.

– *Mechanical swabbing:* The die should be gently brushed with an acetone-soaked lab wipe, which should be lint-free to avoid contaminating the die. The sample is scratched carefully to avoid loosening the bond wires.

10.2.2.2 Delayering

Modern chips are made up of several metal layers, passivation layers, vias, contact, poly, and active layers. Reverse engineers must perform cross-section imaging of a chip, using SEM or TEM to identify the number of layers, metal material, layer thickness, vias, and contacts. The knowledge from cross-sectional imaging is critical (that is, thickness of the layers), as it determines how the delayering must be performed.

Table 10.2 Decapsulation of a die using different methods and their pros and cons

Decapsulation Methods		Pros	Cons
Chemical	Wet	Using sulfuric or nitric acid, it has a high etch rate Works well when die size is small compared to package	Does not work with ceramic packages Acid can damage lead frame, and bond wires Isotropic etch
	Dry	Removes material with good selectivity Can remove any material	Slow for ceramic packages Contamination of etcher may result in uneven removal of material
Mechanical	Grinding and polishing	Even removal of material Easy to use More suitable for flip chips	Works when lead frame is higher than back side of the die Does not work on certain areas
	Milling	Removes material in a specific area Three-axis material removal	Needs professional skills to work with CNC Accuracy of material removal is limited with the tool accuracy
	Thermal shock	Fast and inexpensive process Easy to perform	High risk of damaging die Not controllable in specific areas
Nanoscale fabrication techniques	High-current FIB	High accuracy in material removal (nm) Can be performed on controlled area	Expensive, Requires, high operation skills Slow milling rate (30 nm^3/s)
	Plasma FIB	High accuracy in material removal (nm) Can be performed on controlled area Faster milling rate (2000 nm^3/s)	Expensive Requires high operation skills
Laser ablation		Accurate in material removal (μm) Can be performed on controlled area Faster milling rate (10^6 $\mu m^3/s$)	Expensive Requires high operation skills

Table 10.3 Wet etching recipes for different types of metals and etching process [34]

Material to Be Etched	Chemicals	Ratio	Etching Process and Comments
Aluminum (Al)	H_3PO_4 : Water : Acetic Acid : HNO_3	16:2:1:1	PAN Etch; 200 nm/min @ 25 °C; 600 nm/min @ 40 °C
Aluminum (Al)	NaOH : Water	1:1	May be used @ 25 °C but etches faster at a higher temperature
Silicon (Si)	HF : HNO_3 : Water	2:2:1	–
Copper (Cu)	HNO_3 : Water	5:1	–
Tungsten (W)	HF : HNO_3	1:1	–
Polysilicon (Si)	HNO_3 : Water : HF	50:20:1	Remove oxide first; 540 nm/min @ 25 °C
Polysilicon (Si)	HNO_3 : HF	3:1	Remove oxide first; High etch rate: 4.2 μm/min
Silicon, dioxide, (SiO_2)—thermally grown	HF : Water	1:100	Very slow etch; 1.8 nm/min @ 25 °C
Silicon dioxide, (SiO_2)—thermally grown	HF	–	Very rapid etch; 1.8 nm/min @ 25 °C
Silicon nitride (Si_3N_4)	Refluxing phosphoric acid	–	Use at 180 °C; 6.5 nm/min @ 25 °C; Plasma etching is preferred for removing Si_3N_4

Several methods can be used simultaneously when a chip is delayered, such as wet/plasma etching, grinding, and polishing. A reverse engineer should determine the etchants needed, and the time required to remove each layer, because the layout could depend on the specific technology, which could be either CMOS, or bipolar. For example, memory device vias are much higher than others, so etching is challenging, because one has to remove a large amount of material. Several types of metals and required wet etchants are shown in Table 10.3 [34].

Once the etchants are determined for delayering a specific layer and metal, a reverse engineer will begin by etching the passivation layer; then, the reverse engineer will take an image of the highest metal layer; after that, the reverse engineer will etch the metal layer. This same process is repeated for each layer, including the poly and active layers. When delayering a chip, the layer surface has to be maintained as planar, and one at a time, each layer should be etched carefully and accurately [3,4]. In addition, the layer thickness of a chip could vary because of manufacturing process variations. The best approach is to have one die for every level of delayering. For example, when delayering is done for a four-layer chip, a reverse engineer could use four dies for each metal layer of the chip.

To delayer a chip accurately, an advanced laboratory should have one or more of the following pieces of mechanical equipment [4]: a semi-automated polishing machine, a semi-automated milling machine, a laser, a gel etch, a CNC milling machine, and an ion beam milling machine. When the chip has been delayered, one could face the following challenges [4]:

– *Planarity of the layer:* The planarity of the layer could be conformal or planarized. In a conformal layer, some portion of the different layers and vias could appear on the same plane. However, in a planarized layer, only one layer appears at a time. Conformal layers are more challenging.

– *Material removal rate:* The equipment could be slow or fast and could underetch or overetch.

– *Die size:* Thickness, length, and width can vary.

– *Number of samples:* There may not be enough parts to image each layer separately (that is, information on a layer could be missing if delayering is not done accurately).

– *Selectivity of the material:* One must be careful to remove one material but not another (for example, removing a metal layer without affecting the vias).

10.2.2.3 Imaging

During the delayering process, thousands of high-resolution images are taken to capture all of the information contained in each layer. Later, these images can be stitched together, and then studied to recreate the chip. For the purposes of imaging, many high-resolution microscopes and x-ray machines could be used as discussed in Section 10.2.1.

10.2.2.4 Post-Processing

The post-processing or circuit extraction after delayering consists of the following steps: (1) image processing, (2) annotation, (3) gate-level schematic extraction, (4) schematic analysis and organization, and (5) high-level netlist extraction from the gate-level schematic. Each of these steps is described in greater detail as follows:

Image Processing. Taking images manually is becoming increasingly difficult, because the size of the ICs is shrinking, along with many of their features [3]. Advanced electrical labs now use automated instruments (x-rays, SEMs, digital microscopes) that are equipped to take images of entire layers of ICs and PCBs. Then, the automated software can be used to stitch the images together with minimal error, and synchronize the multiple layers without misalignment. In addition, it is important to establish the lineup of the layers' contacts and vias before the extraction.

Annotation. After the completion of the aligned layers and stitched images, the extraction of the circuit starts. This stage in the process includes making note of transistors, inductors, capacitors, resistors, diodes, other components, the interconnection of the layers, vias, and contacts. The circuit extraction could be an automated or a manual process. For example, Chipworks has an ICWorks extractor tool that can look at all of the imaged layers of the chip and align them for extraction [3]. The tool can be used to view several layers of a chip in multiple windows, simultaneously. The ICWorks extractor tool might also be used for the annotation of wires and devices. Image recognition software (2D or 3D) is used for the recognition of standard cells in digital logic. Automated image recognition software helps to facilitate the extraction of large blocks of digital cells quickly.

Gate-level schematic extraction. Sometimes the images are imperfect, as the images may be taken manually. Additionally, the annotation process and image recognition for digital cells could be erroneous. Therefore, verification is needed before the creation of a schematic. Design rule checks could be used to detect any issues related to minimum-sized features or spaces, wire bonding, vias, and connections [3]. After this stage, tools such as ICWorks can extract an interconnection netlist from which a flat schematic could be created. The schematic could be checked for any floating nodes, shorted input or output, or supplies and nets that have no input or output. The annotations, netlist, and schematic depend on each other, so changing one could affect the others.

Schematic analysis and organization. The schematic analysis should be done thoughtfully and carefully with proper hierarchy and design coherence. For the analysis and organization of a schematic, the reverse engineer could use public information on the device, such as its datasheet, technical report, marketing information, and patents. This could help to facilitate an analysis of the architecture and circuit design. Some structures, such as differential pairs and bandgap references, could be easily recognizable.

High-level netlist extraction from gate-level schematic. After circuit extraction is performed on the stripped IC (derivation of circuit schematic diagram), several techniques [35–37] could be applied to get the high-level description for analysis and validation of the functionality of the chip, using simulation. [35] proposed RE from a gate-level schematic of ISCAS-85 combinational circuits to get the circuit functionality by computing truth tables of small blocks, looking for common library components, looking for structures with repetition, and identifying bus and control signals. [38] presents RE of gate-level netlists to derive the high-level function of circuit components based on behavioral pattern mining. The approach is based on a combination of pattern mining from the simulation traces of the gate-level netlist, and interpreting them for the pattern graph. [38] proposed an automatic way to derive word-level structures that could specify operations from the gate-level netlist of a digital circuit. The functionality of logic blocks is isolated by extracting the word-level information flow of the netlist, while considering the effect of gate sharing. A variety of algorithms are used by [37] to identify the high-level netlist with module boundaries. The algorithms are applied for verification to determine the functionality of components, such as register files, counters, adders, and subtractors.

10.2.3 CHIP-LEVEL ANTI-RE

There are several approaches for the anti-RE of ICs, which include camouflage, obfuscation, and other techniques. Following is a description of these methods:

10.2.3.1 Camouflage

Layout-level techniques such as cell camouflage [39,40] and dummy contacts could be used to hinder adversaries who want to perform RE on a chip. In the camouflage technique, the layout of standard cells with different functionalities is made to appear identical. One can introduce camouflage to a standard gate by using real and dummy contacts, which can enable different functionalities, as shown in Fig. 10.8. In Figs. 10.8A and 10.8B, the layouts of two-input, NAND and NOR gates, are shown. These gates functionalities can be easily identified by their layouts. In contrast, Figs. 10.8C and 10.8D show camouflaged two-input NAND and NOR gates with layouts that appear identical. If regular layouts are used for standard gates, automated image processing techniques can easily identify the functionality of the gates (see Figs. 10.8A and 10.8B). Camouflaging (see Figs. 10.8C and 10.8D) can make it more difficult to perform RE with automated tools. If the functionality of the camouflage gates of the design is not correctly extracted, the adversary will end up with the wrong netlist.

10.2.3.2 Obfuscation

Obfuscation techniques entail making a design or system more complicated to prevent RE, while also allowing the design or system to have the same functionality as the original. There are several different obfuscation approaches in the literature [41,42]. The hardware protection through obfuscation of netlist could be used against piracy and tampering, and the technique could provide protection at every level of the hardware design and manufacturing process [42]. The approach is achieved by obfuscating the functionality by systematically modifying the state-transition function and internal logic structure of the gate-level IP core. The circuit will traverse the obfuscated mode to reach the normal mode only for specific input vectors, which are known as the "key" for the circuit.

[41] proposed a technique of interlocking obfuscation in the register transfer level (RTL) design, which could be unlocked for a specific dynamic path traversal. The circuit has two modes: entry mode

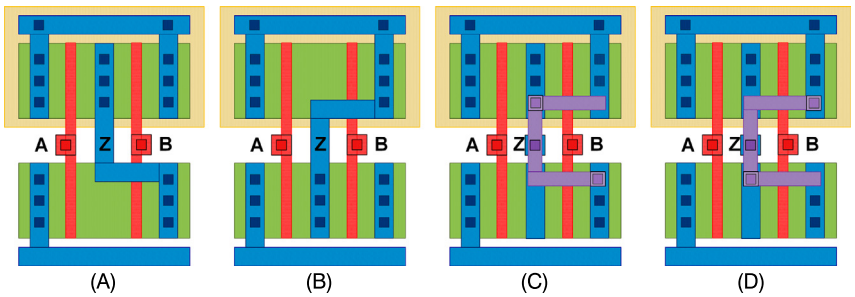

FIGURE 10.8

(A) Standard NAND gate and (B) NOR gate. These gates could be easily differentiable by looking at the top metal layers. (C) Camouflaged NAND gate and (D) NOR gate. These gates have identical top metal layers and are, therefore, harder to identify.

(obfuscated) and functional mode. The functional mode is operational when there is a formation of a specific interlocked code word. The code word is encoded from input to the circuit, which is applied in entry mode to reach the functional mode. This code word is interlocked into the transition functions, and is protected from RE by increasing the interaction with the state machine. The additional benefit is that any minor change or alteration to the circuit made by an adversary will be magnified due to the interlocking obfuscation. The technique has a large area overhead, so there is a trade-off between the area overhead and the level of protection. Higher protection levels require larger overheads.

10.2.3.3 Other Techniques

Today, most companies are fabless, meaning that the fabrication of chips is outsourced. A semiconductor foundry is given the design [43] to fabricate the chips. To accomplish post-fabrication control of the ICs that are produced in such plants, IC hardware metering protocols have been put in place to prevent IC piracy [10,44]. ICs can be identified by active metering, which is a process by which parts of the chip can be used for locking and unlocking by the design house. Physical unclonable functions (PUFs) can be used to generate secret keys to protect from cloning [44,45]. PUF is difficult to duplicate. Therefore, RE and cloning of the whole chip could be possible, but the reverse engineer would not be able to activate the cloned chip.

[11] proposed a reconfigurable logic barrier scheme that separates information flow from the inputs to the outputs. This technique is used in the IC pre-fabrication stage for protection against IC piracy. The information could flow with the correct key, but the barrier would interrupt flow for the incorrect key. The main difference between the logic barrier scheme and the obfuscation techniques, described in Section 10.2.3.2, is that the logic barrier scheme is based on the proper locking locations of the barrier in the design instead of randomized ones. This technique is used for effectively maximizing the barrier with minimum overhead by utilizing better-defined metrics, node positioning, and enhancing the granularity from XOR gates to look-up tables (LUTs).

An external key could be placed in every chip for protection against IC piracy. This method is called end piracy of integrated circuits (EPIC) [15]. This key is produced by the IP holder, and is

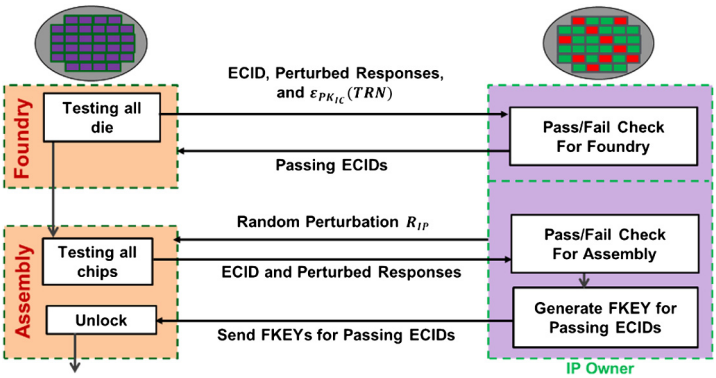

FIGURE 10.9

Overview of the secure split-test (SST).

unique. Manufacturers must send the ID to the IP holder for the chip to become functional, and the IP holder must then send the activation key to enable the activation of the chip with the ID. The random ID is generated by several techniques. This ID is generated before the testing of the IC. This key prevents cloning of the IC from RE, and controls how many chips should be made. The EPIC technique's limitations include complex communication with the IP holder, which could impact test time and time to market. Additionally, this technique requires higher levels of power consumption.

[46] proposed a bus-based IC locking and activation scheme for preventing unauthorized manufacturing. The technique involves the scrambling of the central bus, so that the design can be locked at the manufacturing site as a means of guaranteeing the chip's uniqueness. The central bus is controlled by both reversible bit permutations and substitutions. A true number generator is applied to establish the code for the chip, and the Diffie–Hellman key exchange protocol is employed during activation.

[47] proposed a method called secure split-test (SST) for securing the manufacturing and testing process of SoC, and give control back to the SoC designers to prevent counterfeit, defective and/or out-of-spec SoC from entering the supply chain. In SST, each chip is locked during the testing process. The SoC designer is the only entity who can interpret the locked test results, and unlock the passing chips. In this way, SST can prevent overproduction, and also prevent chips from reaching the supply chain. SST, in addition, establishes unique key for every chip, which drastically improves security against supply chain attacks.

SST consists of the following components: (1) true random number generator (TRNG); (2) fuse-based storage for true random numbers (TRNs); (3) public key encryption/decryption unit; (4) scan-locking module; and (5) functional-locking block. An overview of SST is shown in Fig. 10.9. One can assume that an ECID has been securely generated before starting tests using the mod-EaaS approach developed in Task 1. The test process at the foundry begins with an initialization step, where the TRNG generates a TRN (similar to the protocol in Task 1) and stores it in a nonvolatile memory. This TRN is used to both uniquely perturb test responses, and lock each IC. TRN is shared with the IP owner by encrypting it with a public key PK_IC, hardcoded into the IC design. By knowing TRN, the IP owner can determine if the IC passes tests, and how to generate the key (referred to as FKEY) to unlock the IC. The IP owner will identify which die pass, and send the corresponding ECIDs to the

foundry. The foundry sends only the passing die to the assembly for packaging. The IP owner sends a random number $R_I P$ to the assembly for each IC. The $R_I P$ adds randomness to the process, in case the foundry/assembly collude during the test process. A similar round of communication occurs between assembly and IP owner. The IP owner generates FKEYs and sends them with the associated ECIDs to the assembly only for the ICs that pass testing. The FKEY is burnt into nonvolatile memory of the corresponding IC, thereby unlocking it.

10.2.4 BOARD-LEVEL RE

The goal of board-level RE is to identify all components on the board and the connections between them. All of the components used in a design are called the bill of materials (BOM) [1]. The components and parts of a PCB could be any of the following: microprocessors, microcontrollers, decoupling capacitors, differential pairs, DRAMs, NAND flashes, serial EEPROMs, serial NOR flashes, and crystals/oscillators. There could be silkscreen markings, high-speed serial/parallel ports, program/debug ports, JTAGs, DVIs, HDMIs, SATAs, PCIs, Ethernets, program/debug ports, and display ports [3,48]. To identify the components, test points, and parts of the PCB, silkscreen markings are often used [1]. For example, D101 may be a diode, and Z12 might be a zener diode.

IC identification via chip and die markings. Some electronic components mounted on the PCB can be identified easily through the use of IC markings, but fully custom and semicustom ICs are difficult to identify. Using standard off-the-shelf parts with silkscreen annotations will assist the RE process. If the ICs have no markings, then the manufacturer's logo can give an idea of the functionality of the chip. Custom devices, which are developed in-house, are difficult to identify [1], because a custom device could be undocumented, or documentation could be provided only under a nondisclosure agreement.

IC markings can be divided into the following four parts [49]:

- The first is the prefix, which is the code that is used to identify the manufacturer. It could be a one- to a three-letter code, although a manufacturer might have several prefixes.
- The second part is the device code, which is used to identify a specific IC type.
- The next part is the suffix, which is used to identify the package type and temperature range. Manufacturers modify their suffixes frequently.
- A four-digit code is used for the date, where the first two digits identify the year and the last two identify the number of the week. In addition, manufacturers could cipher the date into a form only known by them.

The marking conventions of a Texas Instruments (TI) chip for the first and second line is shown in Fig. 10.10. The TI chips could have an optional third and fourth line with information related to the trademark and copyright. After identifying the manufacturer and IC markings, the reverse engineer could find the detailed functionality of the chip from the datasheets, which are available on the Internet [50,51].

If the IC marking is not readable, because it has faded away due to prior usage in the field or the manufacturer did not place a marking for security purposes, the reverse engineer could strip off the package, and read the die markings to identify the manufacturer and the chip's functionality [49]. The die marking could help to identify the mask number, part number, date of the die mask completion or copyright registration, company logo, and the trademark symbol. A die marking could match the

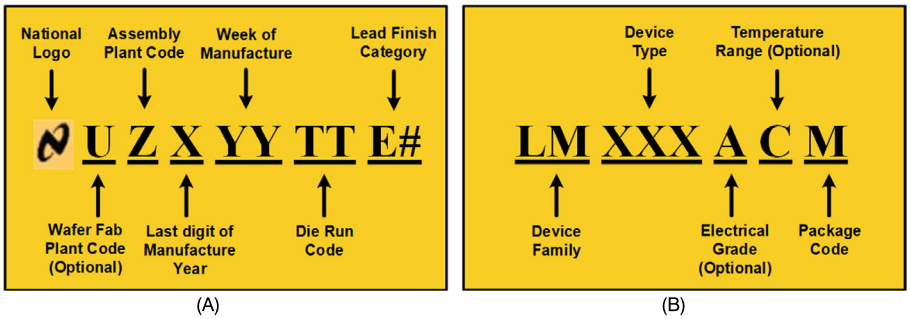

FIGURE 10.10

Marking convention on the TI chips for (A) the first line and (B) the second line.

package marking depending on the manufacturer. Then the datasheet information could be used to assess the die. Die markings are similar within families of chips made by the same manufacturer [52]. So, if someone can find the functionality of one chip, then that person can also identify the functionality of the chip family, because of the almost similar die markings that are shared by the chips in that family. For example, the Qualcomm MSM8255 processor is identical to the MSM7230 in both functionality and design, and both chips are from the Snapdragon family of ICs [52]. The only difference between these two chips is their clock speed. After identifying the components of the PCB, the reverse engineer would want to identify the PCB-type, which could be any of the following: single sided (one copper layer), double sided (two copper layers), or multi-layered. In multi-layered PCBs, chips are connected to each other on the front and the back, and through the internal layers. Some of the internal layers are used as power and ground layers. Conductors of different layers are connected with vias, and delayering is needed to identify these connections.

Destructive analysis of PCBs. Before PCB delayering, images of the placement and orientation of all outer layers' components are captured [1]. Then the components could be removed, drilled hole positions could be observed, and it could be determined whether there are any buried or blind vias. The PCB delayering process is similar to the one described for chips, and therefore will not be discussed further. After the PCB is delayered, images of each layer can be taken [48]. Then the composition and the thickness of the layers should be noted. It is important to track the impedance control of high-speed signals and the characteristics of the PCB. The dielectric constant, prepreg weave thickness, and resin-type should also be determined [1].

Nondestructive 3D imaging of PCBs using x-ray tomography. X-ray tomography is a noninvasive imaging technique that makes it possible to visualize the internal structure of an object without the interference of over-and underlayer structures. The principle of this method is to acquire a stack of 2D images, and then use mathematical algorithms such as the direct Fourier transform and center slice theory [53] to reconstruct the 3D image. These 2D projections are collected from many different angles, depending on the quality needed for the final image. The object properties, such as dimension and material density, source/detector distance to object, source power, detector objective, filter, exposure time, number of projections, center shift, and beam hardening are important to consider in the selection of the tomography process parameters. Internal and external structures will be ready to analyze when

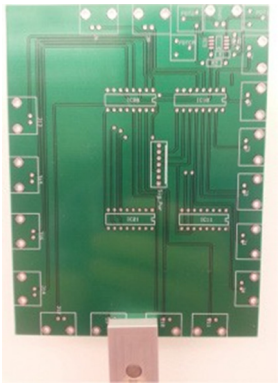

FIGURE 10.11

PCB mounted on a sample holder.

the 3D image is reconstructed [48]. A discussion of how to select the right values for any of these parameters is outside the scope of this article. More information on tomography parameters is available in [54].

As an example, the traces and via holes of a four-layer custom PCB using a Zeiss Versa 510 x-ray machine are analyzed [55]. To make sure that features on the board can be observed, they selected a fine pixel size, which gives us high enough image quality. After several rounds of optimization, the tomography parameters for obtaining the best quality images are selected. The process is completely automated after setting the parameters, can be performed without the need for oversight, and should be widely applicable to most PCBs.

For the four-layer custom board in Fig. 10.11, all traces, connections, and via holes are clearly captured. To validate the effectiveness of the tomography approach, the results are compared to the board design files previously used to produce the PCB. The board includes a front side, back side, and two internal layers. The internal layers correspond to power and ground. The via holes connect the traces on two sides of the board, and are also connected to either power or ground layers. The internal power layer is presented in the design layout in Fig. 10.12.

The 3D image of the board is reconstructed using a combination of thousands of virtual 2D slices. These slices can be viewed and analyzed separately. The thickness of each of these is same as the pixel size (that is, 50 μm). In Fig. 10.13, one slice is provided, which shows the information of the internal power layer.

By comparing the tomography results and the design layout of the board, one can see a clear difference between the via holes that are connected, and those that are not connected to the internal layer. Soldered joints constitute a highly x-ray absorbing material, and result in white contrast for the associated pixels. However, plastic has a lower density and is more x-ray transparent, which results in a dark contrast. Thus, one can easily determine which via holes are connected to an internal layer. The same principle will let us detect the traces on the side layers of the board due to the presence of copper on the traces, as shown in Fig. 10.14.

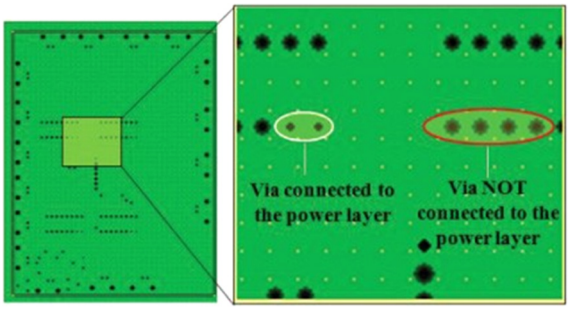

FIGURE 10.12

Layout design of the internal power layer.

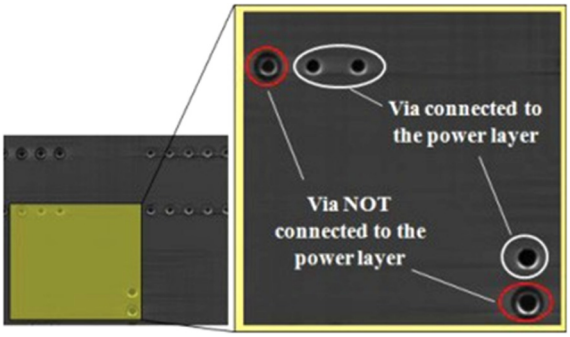

FIGURE 10.13

Virtual slicing presents power layer.

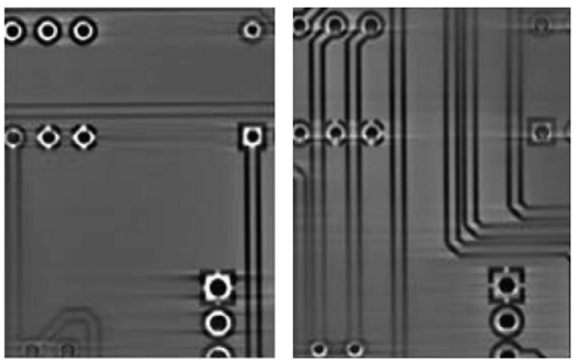

FIGURE 10.14

Reconstructed (A) top and (B) bottom layers of a PCB.

Netlist extraction after imaging. After capturing images of the PCB via delayering or x-ray tomography, connections between all of the components could be discovered, which would yield a PCB layout netlist. Then commercial tools could be used for converting the layout back into schematic [56]. To create the netlist from the collected images, one should verify the following:

- connection between the components of original board (a datasheet could be helpful to find the connection for original functionality),
- unexpected shorts and hanging Vdd,
- pin connections between components.

Several techniques have been used for analyzing x-ray images in prior work [57,60]. Wu et al. [57] uses a visual inspection system for PCBs. The elimination subtraction method is used, which subtracts the perfect PCB image (the template) from the inspected image, and locates the defects in the PCB. Mat et al. [58] applied structuring technique to a raw PCB image (input) using a morphological operation. After that, a dilation and erosion function is applied, so that a fine-segmented image of the PCB tracks can be achieved. Koutsougeras et al. [59] applied an automatic Verilog HDL model generator, which includes the image-processing technique that is used to identify the components and their connections. After that, a circuit graph is obtained, which corresponds to a primitive schematic circuit of the board. Finally, Verilog HDL is generated from the circuit graph. A Verilog XL simulator is used for testing the performance. The layers of the circuit card assemblies/PCBs are separated using x-ray stereo imaging in [60]. The focus is to identify the solder joints and traces on the different layers of a multi-layered PCB. In the automated process technique, photos are taken from one- or two-layer PCBs. Then, a C++ program is used to automatically reverse engineer the netlist.

10.2.5 BOARD-LEVEL ANTI-RE

Ensuring complete protection from PCB-level RE is a difficult task, and thus the goal of anti-RE methods is to simply make RE prohibitively expensive and time consuming. A summary of PCB-level anti-RE techniques follows [1]:

1. Tamper-proof fittings (such as torx), custom screw shapes, adhesively bonded enclosures, and fully potting the space around a PCB could be used for protection against physical attacks.
2. Custom silicon, unmarked ICs, missing silkscreens with minimum passive components, and a lack of information from the Internet could complicate RE. Additionally, the elimination of JTAG and debug ports from silicon can make the RE process harder.
3. Ball grid array (BGA) devices are better, because such devices do not have exposed pins. Back-to-back BGA placement in a PCB board could be most secure, because of the inaccessibility of the unrouted JTAG pins with controlled depth drilling on any side of the PCB. For back-to-back BGA placement, the PCB needs to be multilayered, which will increase the RE cost for layer-by-layer analysis. The problem is that back-to-back BGA packaging is complex and expensive.
4. If the devices are operating in an unusual fashion (for example, if there are jumbled addresses and data buses), then, it would be hard to find the functionality of the device. Obfuscation (that is, wiring connections between used pins to unused pins, having spare inputs and outputs from processors to route signals, dynamically jumbling buses, and jumbling the PCB silkscreen annotations) could

Table 10.4 Implementation challenges of anti-RE techniques for board level

Anti-RE Techniques	Design Cost	Manufacturing Impact	RE Cost
Tamper-proof fittings such as torx and custom screws shapes	Moderate	Low	Very low
Fully potting the space around a PCB	Low	Moderate	Low
Missing silkscreen with minimum passive components	Low	Low	Low
Custom silicon and unmarked IC	Low	Moderate	Low
BGA devices	Low	High	High
Routing, signals for inner, layers only	Moderate	High	Moderate
Multilayer PCB	High	Moderate	Very high
Using blind and buried vias	Moderate	Very high	Moderate
Dynamically jumbled buses	Low	Very low	Low
Route through ASIC	Very High	Moderate	High
Route through FPGA	Moderate	Moderate	Moderate
Elimination of JTAG and debug ports	Low	Moderate	Low

complicate the RE process. However, such techniques also require the use of more complex chips, and complicated design methods.

Many of the preceding methods are difficult to implement and could significantly increase design and manufacturing costs. Table 10.4 shows the effectiveness of anti-RE techniques at the board level [1]. A total of five levels are used for scaling, based on identifying design cost, manufacturing impact, and RE cost.

10.2.6 SYSTEM-LEVEL RE

With chip- and PCB-level RE processes, the purpose is to obtain the netlist of the chip and board in the embedded system, which represents the function and interconnections of the design. To make the design fully functional, the system operation codes and control instructions, which are defined by firmware, should be retrieved as well. This is referred as system-level RE.

Parallel to the embedded system design, involving ASICs and MCU/DSPs, there are designs based on FPGAs, whose share of market has been increasing in modern product design. Considering the fact that the hardware functionality and interconnection (referred to as the netlist) are enclosed in the binary configuration file (called the bitstream), the RE process of FPGA is completely different from the ASIC chip-level RE, which is mainly based on geometrical characteristics of the chip layout (see Section 10.2.2). Here, FPGA RE is categorized into the system-level RE as well, as both the firmware in MCUs, DSPs, and so forth, and netlist information, are stored in the NVM devices.

In this section, first various NVM storage devices are introduced, and then the RE methods used to extract the firmware/netlist accordingly are discussed.

10.2.6.1 Firmware/Netlist Information Representation

Firmware and netlist information can be stored via read-only memory (ROM), electrically erasable programmable ROM (EEPROM), or Flash memory. ROM is a type of memory, whose binary bits are programmed during the manufacturing process. Currently, ROM is still among the most popular

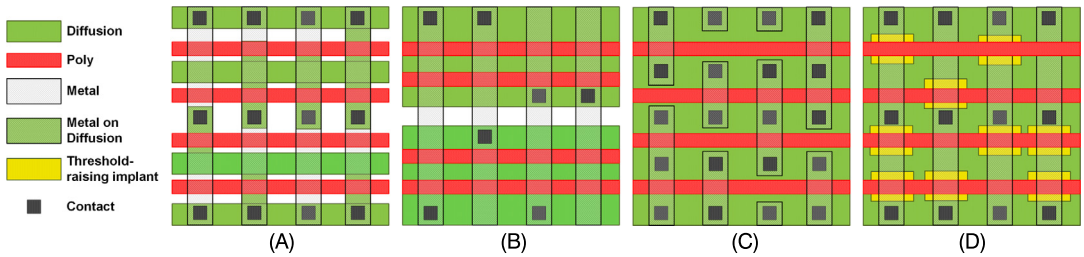

FIGURE 10.15

Illustrations of (A) active-layer programming ROM, (B) contact-layer programming ROM, (C) metal-layer programming ROM, and (D) implant programming ROM.

storage media due to its low cost per cell, high density, and fast access speed. From the perspective of ROM physical implementation, ROM devices can be typically classified into four types [61], as shown in Fig. 10.15:

- Active-layer programming ROM: The logic state is represented by the presence or absence of a transistor. As shown in Fig. 10.15A, a transistor is fabricated by simply bridging polysilicon over the diffusion area.
- Contact-layer programming ROM: A bit is encoded by the presence or absence of a via, which connects the vertical metal bitline with the diffusion area as illustrated in Fig. 10.15B.
- Metal-layer programming ROM: The binary information is encoded by short circuiting the transistor, or not as shown in Fig. 10.15C.
- Implant programming ROM: The different logic state is achieved by different doping levels in the diffusion area (see Fig. 10.15D). Generally, higher doping levels will raise the on/off voltage threshold, which will disable the transistor.

Compared to ROM, EEPROM provides users with the capability to reprogram its content. As shown in Fig. 10.16A, one bit cell of EEPROM is composed of two transistors: floating gate transistor (FGT) and select transistor (ST). The FGT is feathered with two stacked gates: a control gate (CG) and a floating gate (FG). The logic state of the bit cell is encoded in the FGT by the presence or absence of electrons stored in the FG. Being isolated electrically, the FG can retain the electrons when powered off. Flash memory (see Fig. 10.16B) has almost the same structure as EEPROM, except for the absence of ST, which is irrelevant to the logic state, and only allows EEPROM to be byte addressable.

An FPGA bitstream is essentially a vector of bits encoding the netlist information in FPGA, which defines hardware resources usage, interconnection, and initial states at the lowest level of abstraction. The logic blocks are configured to represent the basic digital circuit primitives, such as combinational logic gates and registers. The connection blocks and switch blocks are configured to be the interconnections between different logic blocks. Other hardware resources, such as I/O buffers, embedded RAM, and multipliers, can be programmed according to different requirements. Therefore, all information about the netlist can be obtained from the bitstream file.

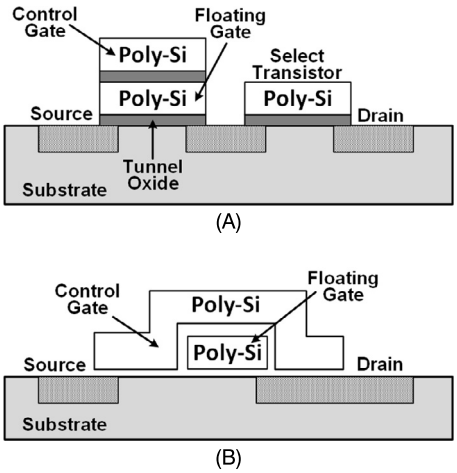

FIGURE 10.16

Illustrations of (A) EEPROM and (B) Flash.

10.2.6.2 ROM RE

To reverse engineer the ROM content, one can take advantage of modern optical and electron microscopy to observe the binary states of each cell, as indicated below:

- Active-layer programming ROM: The metal layer and poly layer need to be removed using the delayering approaches discussed in Section 10.4, for they will obscure the active layer underneath.
- Contact-layer programming ROM: It is much easier to reverse engineer this kind of ROM, as there is often no need to delayer the metal layer and the poly layer. In the relatively old ROM technology, the contact layer is clearly visible, but in more modern technologies, some delaying is still needed to expose the contact layer before observation.
- Metal-layer programming ROM: This type of ROM can be directly observed under a microscope without having to perform any delayering process.
- Implant programming ROM: This type of ROM is inherently resistant to optical microscopy, as different logic states appear identical. To observe the impact of different doping levels, additional dopant-selective crystallographic etch techniques [62] should be utilized to separate the two logic states.

Generally, ROM only provides limited protection against RE. Among all types of ROM, the metal-layer programming ROM offers the worst security, because the metal layer is easy to obtain with little effort, whereas the implant programming ROM provides the highest level of protection available.

10.2.6.3 EEPROM/Flash RE

Since EEPROM and Flash memory have similar structures and the same logic storage mechanism (as discussed earlier), they often can be reverse engineered by the same procedures. Considering that EEPROM/Flash represent different states by electrons, not by the geometric difference, x-ray technology

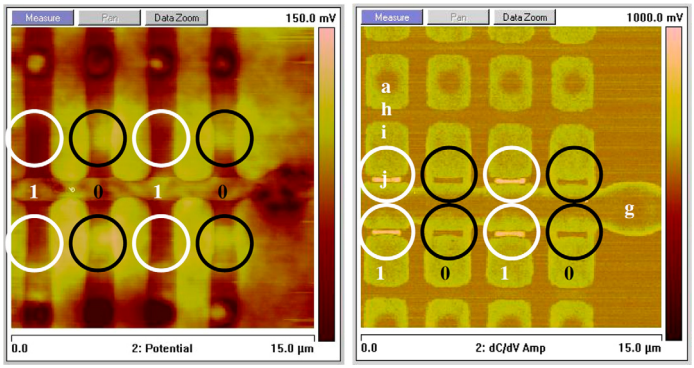

FIGURE 10.17

(A) SKPM scan and (B) SCM scan from the backside of Flash memory [64].

cannot be used to detect the contents. Further, any attempt to delayer and measure the electrons in the FG, such as SEM and TEM, will change the electron distribution, thereby disturbing the content inside.

For quite long time, the EEPROM/Flash technology has been regarded as the most robust memory defense against RE. Recently, several methods [63–65], although very expensive and requiring specialized equipment, were proposed to extract the contents in EEPROM/Flash correctly. Note that both of the following methods are applied from the backside of the memory, as traditional front-side delaying and imaging will cause the charges in the FG to vanish [63].

Scanning Kelvin probe microscopy procedure. The scanning Kelvin probe microscopy (SKPM) procedure [66] directly probes the FG potential through the tunnel oxide layer with a thickness of 10 nm, which isolates the FG with the transistor channel as illustrated in Fig. 10.16A. Thus, the first step is to remove the silicon from the back side of the memory, and leave the tunnel oxide layer undamaged to avoid charging/discharging of the FG. Then, the bit value can be read under the SKPM scan by applying a DC voltage to the probe tip. As shown in Fig. 10.17A, the scanning data from SKPM shows the 2D distributions of potential difference between the tip and the memory cell. The potential difference between the charged FG (associated with "0") and the tip is much higher than that between the uncharged FG (associated with "1") and the tip, which leads to a brighter area for the bit "0" (circled in black in Fig. 10.17A).

Scanning capacitance microscopy (SCM) procedure. Unlike the SKPM procedure, the SCM procedure measures the capacitance variations between the tip (with the sample in the contact mode) and the high-sensitivity capacitance SCM's sensor [67]. Given the fact that the holes will be coupled in the transistor channel with the existing electrons in the FG, the SCM sensor will detect the logic states via probing the carrier (hole) concentration. Thus, the back-side delayering should keep a silicon thickness of 50 to 300 nm to leave the transistor channel undamaged. Then, the bit information can be read as depicted in Fig. 10.17B. The SCM signal shows that the charged FG (associated with "0") has a darker signal (circled in black), which is consistent with high density of holes.

Comparisons between the SKPM procedure and the SCM procedure are summarized in Table 10.5. Note that with technology scaling, the electrons stored in the FG have been reduced to fewer than 1000

Table 10.5 Comparison between SKPM and SCM procedures

Property	SKPM Procedure	SCM Procedure
Delayering position	Back side	Back side
Delayering depth	Entire silicon	50–300 nm thickness
Sensitivity	Low	High
Measured carriers	Electrons	Holes
Measured parameter	Potential	Capacitance
Operation mode	Noncontact	Contact
Application	All EEPROM and some Flash	All EEPROM and Flash

electrons for 90nm-node NAND Flash [64]. In this case, the SKPM procedure can no longer recognize two logic states accurately, whereas the SCM still performs well.

10.2.6.4 RE of FPGAs

FPGA RE involves analyzing the configuration bitstream file and transforming the bitstream file into the hardware netlist, which consists of all components and interconnections at the RTL. To fulfill this goal, hackers need to go through the following steps: get access to the bitstream file from the Flash memory, decrypt the bitstream (if encrypted), and finally build the mapping relationship between the bitstream file and the netlist.

Bitstream Access. SRAM-based FPGA stores the logic cells states in the SRAM, which cannot retain the data after power loss. Therefore, an external NVM device (typically Flash) is adopted to hold the configuration bitstream file and transfer the bitstream file at system boot-up to initiate the SRAM in FPGA. The separation between the bitstream file and FPGA makes it easy to dump the contents of the bitstream file. By using a logic analyzer, one can easily wiretap the JTAG data and command lines to capture the communication between the FPGA and Flash memory during startup.

Bitstream Decryption. To increase the security level of FPGA, most FPGA manufacturers encrypts the bitstream file before storing it in the Flash memory with the encryption standards, such as triple data encryption standard (DES) and advanced encryption standard (AES) [68]. Now the wiretapped encrypted bitstreams will not yield any information for RE, as long as the cryptographic key remains hidden inside the FPGA.

The bitstream decryption process in FPGA RE depends entirely on the attacker's ability to discover the key. Typically, the keys are stored in the embedded NVM by programming the FPGA before loading the encrypted bitstream into FPGA. The invasive and destructive attacks to find out the cryptographic key are usually infeasible, as they will trigger tamper detection in the FPGA to zeroize the secret keys. Thus far, no public report exists on a successful invasive attack toward SRAM-based FPGA.

Recently, it has been reported that the bitstream encryption of several mainstream FPGA series [69–71] is vulnerable to the side channel attacks (SCAs) [72]. Basically, an SCA is a noninvasive attack to exploit the relationship between physical information (power, timing, and electromagnetic emanation) and certain hardware operations in the FPGA implementation. The triple DES encrypted bitstream file from the Xilinx VirtexII Pro FPGA was successfully cracked by the SCA the first time in [69]. The leaked timing and power consumption information is collected when the encrypted bitstream is decrypted by the dedicated hardware engine within the FPGA. By analyzing the collected power consumption and timing behavior, the hypothetical structure of the internal triple DES module can be

verified. Finally, the divide-and-conquer approach is applied to guess and verify a small portion of the key (for example, six bits for triple DES), which reduces the computation's complexity. This process is repeated until the entire key is obtained. The more recent Xilinx FPGAs (Virtex-4 and Virtex-5), which employ a more advanced encryption module (AES-256), have been cracked in [70] by a more sophisticated type of correlation power analysis [73].

In a similar way, the FPGA power consumption or electromagnetic radiation is measured, while the decryption block is operating in the FPGA. More recently, the cryptographic keys in the Altera's Stratix II and Stratix III FPGA families have also been revealed by the same SCA [71]. The fact that all of the preceding attacks can be conducted within several hours reveals the vulnerability of the bitstream encryption.

Bitstream Reversal. Prior to converting the bitstream file into the corresponding hardware netlist, one needs to understand the bitstream structure, which is usually documented by FPGA vendors, and is accessible online. Typically, a bitstream file consists of four parts [74]: command header, configuration payload, command footer, and startup sequence. In the case of the Xilinx FPGA, the configuration payload determines the configuration points (such as LUT, memory, register, and multiplexer) and the programmable interconnection points (switch box). The goal of the bitstream reversal is to find out the mapping relationship between the configuration payload with the configuration points, and the programmable interconnection points. However, this mapping relationship is proprietary and undocumented, which makes the bitstream file itself serve as an obfuscated design to protect the hardware netlist. In the past decade, there have been several attempts to achieve a bitstream reversal.

- *Partial bitstream reversal.* This kind of bitstream reversal only focuses on extracting some specific configurable blocks in FPGA, such as the LUT, configurable logic block, and multiplier from the bitstream file. [75] shows the possibility to identify the embedded IP cores by extracting the contents of the LUT in the Xilinx Virtex-II FPGA.
- *Full bitstream reversal.* [76] makes the first public attempt to convert the bitstream file into the netlist. The set-theoretic algorithm and cross-correlation algorithm [76] were used to build a database linking the bitstream bits to the associated resources (configuration points and programmable interconnect points) in the FPGA. Then, the database is utilized to produce the desired netlist based on any given bitstream file in Xilinx Virtex-II, Virtex-4 LXT, and Virtex-5 LXT FPGAs. This method, however, cannot fully create the netlist, as it only relies on the information from the accessible Xilinx design language (XDL) file, generated from the Xilinx EDA tool, which only provides information on the active configurable resources. The missing information on the static, unused configurable resources in the FPGA, places it some distance away from full bitstream reversal. In [77], XDLRC (Xilinx design language report), a more detailed file generated from Xilinx EDA tool, is used to enhance the creation of the mapping database. Unlike XDL, the XDLRC file can offer all of the information available about active and static configurable resources. However, the test results in [77] indicate new issues that the cross-correlation algorithm cannot perfectly relate all resources in the FPGA with the bits in the bitstream file. Therefore, due to the absence of well-developed bitstream reversal technique, the FPGA embedded system is more robust against RE, compared to ASIC designs and microcontroller designs.

10.2.7 SYSTEM-LEVEL ANTI-RE

In this section, the solutions to increase the cost of RE on firmware and FPGA bitstreams are analyzed and discussed.

10.2.7.1 Anti-RE for ROMs

The most effective solution for increasing the complexity and difficulty of RE against ROM is to use the camouflage method. Simply speaking, the designer makes all of the memory cells identical under optical inspection, no matter what the contents. This type of solution, although increases the costs of manufacture, will force the attacker to spend considerably more time, money, and effort to get access to the ROM contents. Recall that for the implant programming ROM in Section 10.2.6.1, the use of different doping levels to encode information constitutes one kind of camouflage technique. Several other camouflage techniques are provided next.

Camouflage Contacts. Different from the contact-layer programming ROM (see Fig. 10.15B), where the absence or presence of contact will expose the logic states, the camouflage contacts act as false connections between the metal layer and active layer to make the true contacts and the false contacts indistinguishable under optical microscopy [78]. To decode the contents, careful chemical etching has to be applied to find the real contacts, and this is time consuming. From the viewpoint of time/cost, this technique will also increase production periods, and lower the manufacturing yield.

Camouflage Transistors. To improve the security of active-layer programming ROM (see Fig. 10.15A), false transistors are made to confuse the RE attempts, instead of using the absence of transistors [79]. The false transistors, essentially with no electrical functions, have the same top-down view as the true transistors under an optical microscope. To crack the information, the attackers have to use more advanced electrical microscopes to analyze the top view and even the cross-sectional view of the ROM, which is usually economically prohibitive. This kind of design will definitely increase the difficulty of RE on the large scale, whereas it only requires minimal effort during manufacturing.

Camouflage Nanowires. Through the use of nano material, ROM cells are fabricated within the vertical connections between the bit lines and the word lines of a ROM array [80]. The real connections between bit lines and word lines act as transistors, whereas the nonelectrical dummy connections only play the role of design camouflage. Due to the small dimensions of the nanowires, the tiny differences between the dummy connections and real connections are indiscernible, even under advanced electrical microscopy. The major challenge with camouflage nanowires, however, is to manufacture the ROM at high enough volume and yield.

Practically all of the preceding camouflage techniques only need to be adopted on a portion of the whole ROM. To develop a stronger anti-RE ROM, more than one anti-RE technique can be used at once.

Antifuse one-time programming. Admittedly, traditional ROMs are inherently vulnerable to RE procedures. Even ROMs equipped with auxiliary anti-RE designs can only offer limited protection against destructive and invasive RE, though they make the design and fabrication process much more complicated. Currently, ROM replacements (example, antifuse one-time programming (AF-OTP) memory devices) are gaining considerable interest.

The AF-OTP memory exploits, whether the gate oxide is in breakdown or is intact to indicate two logic states. Gate oxide breakdown is achieved after fabrication by applying high voltage to the gate of the transistor. Among several proposed structures [81–83], the split channel 1T transistor antifuse [83] exhibits many advantages over the conventional ROM, with respect to cell area, access speed, and

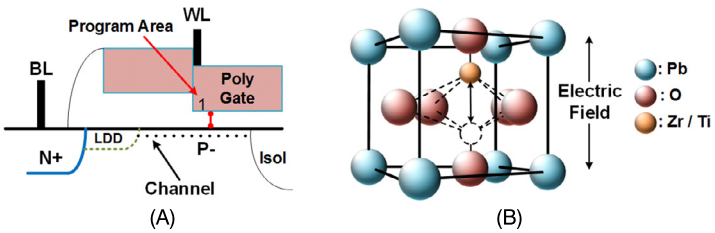

FIGURE 10.18

Illustrations of one memory cell antifuse-OTP (A) and ferroelectric RAM (FeRAM) (B).

immunity to RE. As shown in Fig. 10.18A, the antifuse transistor acts like a capacitor when unprogrammed, but a conductive path will be formed once the oxide is ruptured following the programming operation. Due to the angstrom-level difference between the programmed and unprogrammed antifuse, existing RE techniques (for example, delayering from either the front side or back side, FIB-based voltage contrast [84], and top-down view or cross-sectional view from electrical microscopy) will not expose any information contained, not to mention the fact that it is difficult to locate the oxide breakdown. Additionally, the antifuse memory is compatible with the standard CMOS technology, and thus no additional masks or processing steps are required for fabrication. Considering the security, performance, and cost, the antifuse memory may eventually replace current ROM devices with the feature size continuously scaling down [82].

10.2.7.2 Anti-RE for EEPROMs/Flashs

To reverse engineer the EEPROM/Flash memory, attackers prefer to delayer from the back side to avoid disturbing the floating charges. Thus, the most effective countermeasure would be to prevent back-side attacks. Here, some back-side attack detection methods are briefly introduced, and then one alternative to EEPROM/Flash is reviewed, which can inherently tolerate the back-side attacks.

Circuit parameter sensing. Performing the delayering process from the back side will thin the bulk silicon. By burying two parallel plates in the bulk silicon to form a capacitor, the capacitance sensing [85] can detect the capacitance reduction when the attacker polishes from the back side. When the capacitance reaches below a certain threshold, it will trigger the EEPROM/Flash memory to activate an erase operation. The capacitor, perpendicular to the bulk silicon, was previously a challenge to achieve. Fortunately, the emergence of the through-silicon via technique [86] makes it much easier to fabricate. Similarly, other parameters, such as resistance [87], can be measured and compared to the predefined reference resistance threshold.

Light Sensing. By optically monitoring the back side of the chip, the light-sensing method will equip at least one pair of light-emitting and light-sensing devices in the front side of chip, and light reflection module at the bottom of the silicon bulk [88]. The light-emitting device is configured to emit light, which can penetrate the bulk, be reflected by the light reflection module, and then be collected by the light-sensing device. Once the delayering is applied, the changes in light distribution at the light-sensing device can trigger the self-destruction of the data contained in the memory. This method can certainly make the RE process more time consuming. However, the costs associated with manufacturing and the power consumption from continuous light emitting, and sensing, make it less attractive in practice.

It is worth mentioning that once the detection signal generated from the preceding sensing methods is activated, the memory will automatically erase all or part of its contents. This policy, however, will not cause too much trouble for the RE attackers. For example, the attack can either isolate the charge pump, which provides the power to erase, or ground the detection signal by using a FIB to eventually render all detection-erasure methods useless. In addition, even if the memory successfully erases all of the contents, the attackers still have the chance to determine the actual values according to the residual electrons on the FG due to data remanence [89].

Ferroelectric RAM memory. As mentioned previously, the use of electrons on FGs to represent the logic states makes the EEPROM/Flash memory vulnerable to RE. Recently, ferroelectric RAM (FeRAM) has been shown to be a promising candidate for replacing EEPROM/Flash memory. The motive for FeRAM development is to substantially shorten write time, and lower write power consumption. Recently, it was reported that FeRAM can still possess very strong protections for the contained state [90].

Distinct from the EEPROM/Flash storage mechanism, FeRAM stores data by the polarization states of molecules. These kinds of molecules, located at the middle layer of an FeRAM cell, are capacitors filled with a ferroelectric crystalline material, usually a lead-zirconium-titanate (PZT or $Pb(ZrTi)O_3$) compound. As shown in Fig. 10.18B, the two polarization states, simply the shift up/down of Zr/Ti atom in PZT, represent two different logic states. Due to the high dielectric constant of PZT, the states remain, and only flip under the external electric field.

Due to the special state representations, the difference between two states under optical and electrical inspection is invisible. This is because the distance of the shift up/down (see Fig. 10.18B) is in the scale of nanometer, thereby exposing nothing to the top-down view. One possible attack to reveal the contents, although economically prohibitive, is to carefully slice and analyze the cross-sectional view under SKPM/SCM cell by cell to inspect the difference between the two states.

10.2.7.3 Anti-RE for FPGAs

The fact that the encrypted SRAM FPGA can provide enough RE resilience leaves less space for the research and development of anti-RE techniques compared to the ASIC design. Nevertheless, the existing FPGA anti-RE techniques are categorized into three groups according to the FPGA RE procedure (bitstream hiding, side-channel resistance, and bitstream antireversal). Following is a description of each of these groups:

Bitstream hiding. By integrating the bitstream storage memory with FPGA, the Flash FPGA and antifuse FPGA [91] do not require external configuration memory, leaving the direct wiretapping useless. Unlike the SRAM FPGA, the Flash FPGA does not need bitstream download during powerup, due to the Flash memory nonvolatility. The antifuse FPGA has been widely used in military applications because of its higher RE resilience. As discussed in Sections 10.2.6.3 and 10.2.7.1, an attempt to delayer the Flash memory and antifuse memory, let alone the Flash FPGA and antifuse FPGA, to read out the memory contents, is quite challenging and requires specialized equipment. Although these FPGAs require more fabrication steps than SRAM FPGA, and lack enough programmability due to limited writing times of the Flash/antifuse memories, they are becoming the dominant choice in critical applications.

Side-channel resistance. The recent success of SCAs on the FPGA proves that the leakage of information poses a large threat to FPGA security. Thus, it is necessary to develop the side-channel resistance designs to protect the cryptographic keys. Intuitively, the most effective side-channel resis-

Table 10.6 Costs of anti-RE techniques and RE for system level

Anti-RE Techniques		Anti-RE Cost	RE Cost	Yield Loss
ROM	Camouflage contacts	High	Moderate	Low
	Camouflage transistors	Low	High	Moderate
	Camouflage nanowires	High	High	High
	AF-OTP	Low	Very high	Very low
EEPROM/Flash	Circuit parameter sensing	Moderate	Low	Moderate
	Light sensing	High	Low	Moderate
	FeRAM memory	Moderate	Very high	Very low
FPGA	Bitstream hiding	Very low	High	–
	Side-channel resistance	Moderate	High	–
	Bitstream antireversal	Low	High	–

tance design is to remove the dependency between deciphering operations and power consumption. Tiris et al. [92] presented a dynamic and differential CMOS logic implementation. This technique utilizes a constant power consumption and circuit delay, irrespective of different circuit operations. Wu et al. [93] proposed to adopt the asynchronous logic design to obtain power consumption independent of computations and data. These methods, although effective against SCA, lead to much larger area and power consumptions compared to the standard CMOS logic.

Another group of side-channel resistance designs can be found in the noise addition group. By introducing random power noise to make the power consumption of decryption nondeterministic, it is quite difficult for the attacker to determine which part of the power consumption is from the decryption. Again, this kind of method will introduce new power consumption. In [94], the power reduction technique is proposed to lower the power consumption overhead from noise generation.

Bitstream antireversal. Until now, full bitstream reversal has only been theoretically possible. As one can imagine, the invasive attacks in the future may successfully find out the entire mapping between the encoding bits from the bitstream file and the hardware resources in the FPGA. FPGA vendors should study potential countermeasures to impede bitstream reversal under noninvasive attacks. Currently, bitstream reversal strongly depends on the amount of publicly available information (for example, user guides) and undocumented information (for instance, files generated by EDA tools). It would do well for FPGA vendors to take the possibility of RE attacks into account when releasing new information to hinder potential bitstream reversal attempts.

Another consideration is partial configuration. The critical configuration bits in the bitstream file (such as the IP core) are stored in the Flash memory within the FPGA, whereas other noncritical parts are still loaded from the external memory. This partial configuration only leaves the wiretapper partial information about the whole FPGA mapping information, thereby fundamentally eliminating the potential of bitstream reversal.

10.2.7.4 Summary of Anti-RE Techniques for System Level

Table 10.6 illustrates the cost and the associated yield loss of the system-level anti-RE techniques discussed next. To assess the feasibility of the anti-RE techniques, the costs of RE/anti-RE are classified into five levels based on the previous discussions: very low, low, moderate, high, and very high. It is

worth mentioning that the costs of anti-RE techniques mainly consist of the design and manufacturing costs, whereas the yield loss is estimated from the manufacturing perspective. Other factors, such as power, area, and reliability are not included for lack of open literature. Note also that Table 10.6 only reflects present RE/anti-RE costs. With more effective RE/anti-RE techniques emerging in the future, both RE and anti-RE costs will vary accordingly. In practice, the techniques with lower costs for anti-RE, but higher costs for RE, in Table 10.6 will be more preferably accepted. For ROM, the best choice is clearly antifuse OTP, which has low anti-RE costs, but makes RE very challenging. For EEPROM/Flash, the options are limited, but FeRAM appears to be the most promising. Finally, for FPGAs, bitstream hiding stands out as the best candidate.

10.3 PROBING ATTACK

Physical attacks are capable of bypassing the confidentiality and integrity provided by modern cryptography through observation of a chip's silicon implementation. Such attacks are especially threatening to the integrated circuits (ICs) in smartcards, smartphones, military systems, and financial systems, which process sensitive information. Unlike noninvasive side channel analysis (for example, power or timing analysis), probing directly accesses the internal wires of a security-critical module and extracts sensitive information in electronic format. Probing, in unison with reverse engineering and circuit edit, poses a serious threat to mission-critical applications, and thus demands development of effective countermeasures from the research community [95].

Probing attacks are already a part of the current reality. The most recent example of it emerged when FBI requested help in defeating the passcode retry counter of the Apple iPhone 5c owned by a terrorist suspect. Researchers reverse engineered the proprietary protocol used by the phone's NAND flash, mirrored (copied) the contents, and then brute-forced the passcode in less than a day [96]. While in this case the attack was conducted by researchers, compromise of military technologies through probing could have catastrophic consequences that cost lives. In such instances, advanced IC failure analysis and debug tools are used to internally probe the ICs. Among such tools, focused ion beam (FIB) is the most dangerous.

FIBs use ions at high beam currents for site-specific milling and material removal. The same ions can also be injected close to a surface for material deposition. These capabilities allow FIBs to cut or add traces to the substrate within a chip, thereby enabling them to redirect signals, modify trace paths, and add/remove circuits. Though FIB was initially designed for failure analysis, a skilled attacker can use it to obtain on-chip keys, establish privileged access to memory, obtain device configuration, and/or inject faults. This can be accomplished by rerouting them to an existing output pin, creating a new contact for probing, or reenabling IC test mode. Most of these techniques would not be possible without a FIB. While countermeasures against probing, such as active meshes, optical sensors, and analog sensors have been proposed, they are clumsy, expensive, and ad-hoc. It has been shown time and again that an experienced FIB operator can easily bypass them via circuit edit. In [97], well-known hacker Christopher Tarnovsky probed the firmware of the Infineon SLE 66CX680P/PE security/smart chip from the frontside (that is, top metal layer) by rewiring its active mesh, and making contact with its buses using FIB.

FIB-assisted probing attacks are expected to increase for a variety of reasons. FIBs are becoming cheaper and easier to access than ever before (for example, FIB time can be purchased for a

couple hundred dollars per hour). Further, as FIB capabilities continue to improve for failure analysis, more powerful attacks will be enabled. In contrast, non-invasive and semi-invasive attacks either do not scale to modern semiconductors with Moore's law, or can be mitigated by inexpensive countermeasures. As non-invasive and semi-invasive attacks continue to become less effective, one can expect attackers to migrate to FIB. For these reasons, it is of the utmost importance to stay ahead of attackers and develop more effective countermeasures against FIB-based probing. Since FIB capabilities are almost limitless, the best approaches should make probing as costly, time consuming, and frustrating as possible. A significant challenge in doing so lies in the fact that the time, effort, and cost to design a FIB-resistant chip must remain reasonable, especially to design engineers who are generally not security experts. This could be especially important in the upcoming internet-of-things (IoT) era, which will likely consist of an abundance of low-end chips that are easily physically accessed.

In this section, the state-of-the-art research in the field of circuit edit and anti-probing is presented, the challenges are highlighted, and future research directions for CAD and test communities are offered. The rest of the chapter is organized as follows: Section 10.3.1 reviews technical background related to probing attacks and Section 10.3.2 introduces existing countermeasures against probing attacks and their limitations.

10.3.1 PROBING ATTACK FUNDAMENTALS

Comprehension of the adversary's goal and the techniques he/she uses to successfully carry out probing is the first step in overcoming this significant threat. In this section, technical details of the probing process are reviewed, and associations between technical requirements, decisions, and perceived limitations of state-of-the-art techniques are made.

10.3.1.1 Probing Attack Targets

It is essential for both attackers and countermeasure designers to determine which signals are more likely to be targeted in a probing attack. Such signals are termed as *assets*. An asset is a resource of value, which is worth protecting from an adversary [98]. Unfortunately, a more palpable definition of asset has not been proposed or agreed upon. To help illustrate the wide range of possible information that could be assets, here a few quintessential examples that are the most likely targets for probing attacks are enumerated.

Keys: Keys of an encryption module (for example, private key of a public key algorithm) are archetypal assets. They are usually stored in nonvolatile memory on the chip. If the key is leaked, the root of trust it provides will become compromised, and could serve as a gateway to more serious attacks. An example is original equipment manufacturer (OEM) keys that are used to grant legitimate access to a product, or chip. Leakage of such keys will result in tremendous loss of revenue for the product owner, denial of service, or information leakage.

Firmware and configuration bitstream: Electronic intellectual properties (IPs), such as low-level program instruction sets, manufacturer firmware, and FPGA configuration bitstreams are often sensitive, mission critical, and/or contain trade secrets of the IP owner. Once compromised, counterfeiting, cloning, or exploits of system vulnerabilities could be facilitated.

On-device protected data: Sensitive data, such as health and personal identifiable information, should be kept private. Leakage of such information could result in fraud, embarrassment, or property/brand damage for the data owner.

Device configuration: Device configuration data control the access permissions to the device. They specify which services or resources can be accessed by each individual user. If the configurations are tampered with, an attacker could illegally gain access to resources to which, otherwise, he/she had no access.

Cryptographic random number: Hardware generated random numbers, such as keys, nonces, one-time pads, and initialization vectors for cryptographic primitives also require protection. Compromising this type of asset will weaken the cryptographic strength of the digital services on the device.

10.3.1.2 Essential Technologies for a Probing Attack

A successful probing attack entails a time-consuming and sophisticated process. Countermeasure designers are often interested in ways to make this process go astray. For this purpose, the central approaches and technologies used in published attacks in the following sections are examined.

Front-side vs. back-side: Probing attack targets are those metal wires that carry assets, henceforth called target wires. The most common approach to reach target wires is to expose them from the back end of line (BEOL), that is, from the top metal layer towards silicon substrate (illustrated in Fig. 10.19A). This is called a front-side probing attack. Exposure of target wires is first facilitated with FIB milling. Then, an electric connection to the target wire can be established, for example, by conductor deposition capability of the FIB. Finally, extraction of sensitive information ensues.

A back-side probing attack, that is, probing that occurs through the silicon substrate, was proposed in [99]. Back-side attack targets are not limited to wires. By exploiting a phenomenon during transistor activity, known as photon emission, transistors can also be probed to extract information.

Electrical probing vs. optical probing: The method to access assets shown in Fig. 10.19A is typical for electrical probing, that is, accessing an asset carrying signal via electrical connection. A different approach is optical probing, as shown in Fig. 10.19B. Optical probing techniques are often used in back-side probing to capture photon-emission phenomena during transistor switching. When transistors are switching, they spontaneously emit photons without external stimuli. By passively receiving and analyzing the photons emitted from a specific transistor, the signal processed by that transistor can be inferred. Compared to electrical probing, the optical approach has the advantage of being a purely passive observation, which makes it very difficult to detect. In addition to photon emission analysis, laser voltage technique (LVX), or electro-optical frequency modulation (EOFM), are also used during back-side attacks. These techniques actively illuminate the switching transistors, and then infer asset signal values by observing the reflected light.

The primary deficiency of optical probing lies in the fact that photons emitted in these techniques are infrared, due to silicon energy band gap, which has a wavelength of 900 nm or higher [99]. Therefore, the optical resolution between transistors is limited to within one order of magnitude of the wavelength, due to Rayleigh criterion.

10.3.1.3 Essential Steps of a Probing Attack

In this subsection, the examination of probing attack fundamentals are continued by outlining its essential steps.

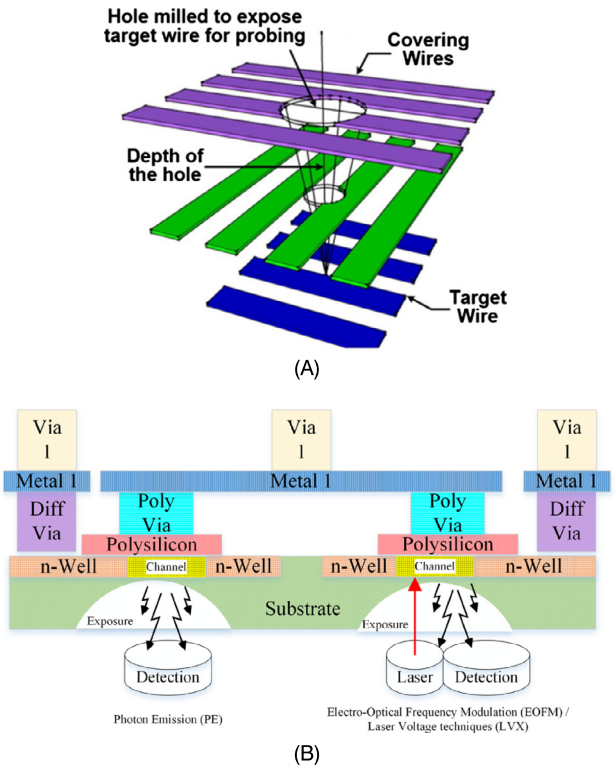

FIGURE 10.19

(A) Milling from BEOL through covering wires (purple and green [medium and light gray in print version, respectively]) to reach target wires (blue [dark gray in print version]); (B) Optical probing: photon emission (PE) and electro-optical frequency modulation (EOFM), or laser voltage techniques (LVX) are used for passive and active measurements, respectively.

Decapsulation: The first stage of most invasive physical attacks is to either partially or fully remove the chip package in order to expose the silicon die. This requires adequate practice and expertise in handling harmful chemicals. Acid solutions, such as fuming nitric acid combined with acetone at 60 °C, are often used to remove plastic packages [100]. Decapsulation can also be done from the back-side of the chip by removing the copper plate mechanically, without chemical etching.

Reverse Engineering: Reverse engineering [55] is the process of extracting design information from something, typically to reproduce it. In the case of probing, reverse engineering is used to understand how the chip works, which requires that the layout and netlist be extracted. By studying the netlist, the attacker can identify the assets. One-to-one correspondence between the netlist and layout can then determine the locations of target wires and buses; and in the event where cutting off a wire is unavoidable, determining whether the cut would impact asset extraction. State-of-the-art tools, such as ICWorks from Chipworks, can perform automatic extraction of netlists from images of each layer

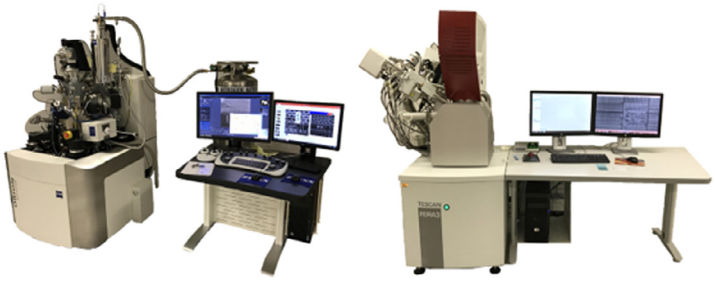

FIGURE 10.20

(A) Scanning electron microscope (SEM); (B) Focused ion beam (FIB). Note that attacker does not need to purchase all these instruments since rent by time is quite low-cost.

taken with optical or scanning electron microscopes (SEM in Fig. 10.20A), which greatly reduces the attacker's effort.

Locating target wires: Once the probing wire targets have been identified by reverse engineering, the next stage is locating the wires associated with the target on the IC under attack. The crux of the problem here is that while the attacker has located target wires on sacrificial devices during reverse engineering process, he/she now has to find the absolute coordinates of the point to mill blindly. This requires a precise-enough kinematic mount, and fiducial markers (that is, visual points of reference on the device) to base these absolute coordinates.

Reaching target wire and extracting information: With the help of modern circuit editing tools like FIB (see Fig. 10.20B), a hole can be milled to expose the target wire. State-of-the-art FIBs can remove and deposit material with nanometer resolution, which allows an attacker with a FIB to edit out obstructing circuitry, or deposit conducting paths that may serve as electrical probe contacts. This feature indicates that many countermeasures can be disabled by simply disconnecting a few wires, and that a FIB-equipped attacker could field as many concurrent probes as logic analyzer allows. Once a target wire is exposed—assuming it is contacted without triggering any probing alarm signals from active or analog shields—the asset signals need to be extracted, for example, with a probe station. The difficulty of this step depends on a few factors. First, software and hardware processes might need to be completed before the asset is available. Further, the sensitive information may not be in the same clock cycle. If the chip has an internal clock source to prevent external manipulation, the attacker will need to either disable it, or synchronize his own clock with it.

10.3.2 EXISTING COUNTERMEASURES AND LIMITATIONS

In the past decade, researchers have proposed various technologies to protect security-critical circuits against probing attacks. In this section, a few representative countermeasures are reviewed and their limitations are highlighted. Unfortunately, to date, none of them offer a satisfactory solution. Further, no method has been proposed to adequately address back-side probing attacks.

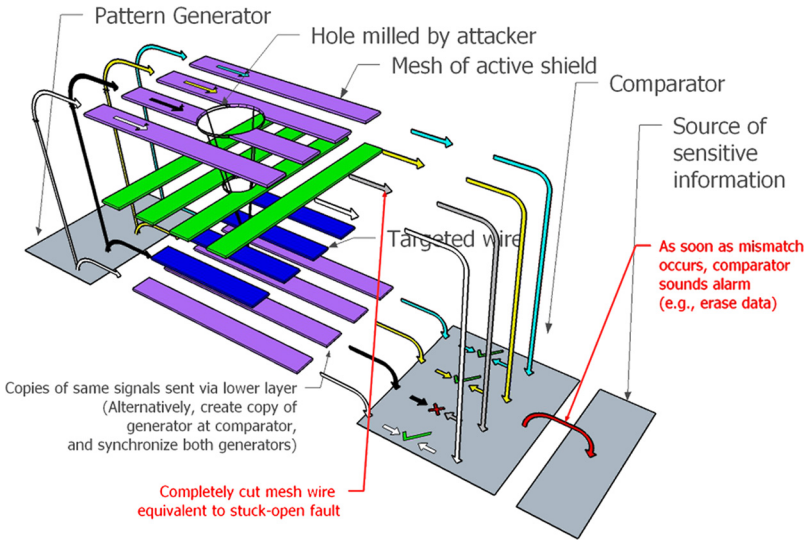

FIGURE 10.21

Basic working principle of active shields.

10.3.2.1 Active Shields

Active shield is, so far, the most investigated probing countermeasure. In this approach, a shield which carries signals is placed on the top-most metal layer to detect holes milled by FIB. The shield is referred to as "active" because signals on these top layer wires are constantly monitored to detect if milling has cut them [101]. Figure 10.21 shows one illustrative example. As shown in the figure, a digital pattern is generated from a pattern generator, transmitted through the shield wires on top-most metal layer, and then compared with a copy of itself transmitted from lower layer. If an attacker mills through the shield wires on top layer to reach target wire, the hole is expected to cut open one or more shield wires, thereby leading to a mismatch at the comparator and triggering an alarm signal to erase or stop generating sensitive information. Despite its popularity, active shields are not without shortcomings. Their biggest problems are that they impose large overheads on the design, but at the same time are very vulnerable to attacks with advanced FIBs, for example, circuit editing attacks.

10.3.2.2 Analog Shields and Sensors

An alternative approach to active shield is to construct an analog shield. Instead of generating, transmitting, and comparing digital patterns, analog shields monitor parametric disturbances with its mesh wires.

In addition to shield designs, the probe attempt detector (PAD) [102] (shown in Fig. 10.22) also uses capacitance measurement on selected security critical wires to detect additional capacitance introduced by a metal probe. Compared to active shields, analog shields detect probing without test patterns and require less area overhead. The PAD technique is also unique in remaining effective against electrical

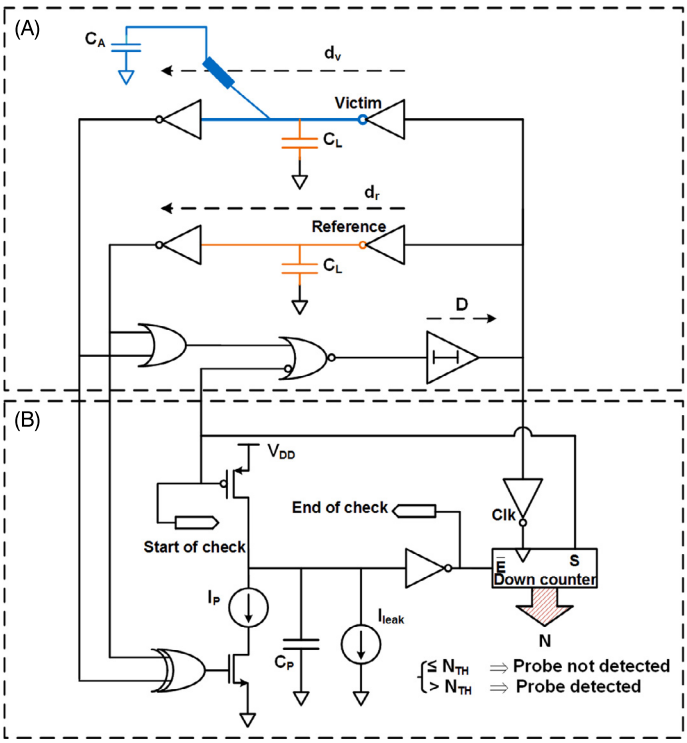

FIGURE 10.22

Probing attempt detector (PAD).

probing from the back-side. The problem with analog sensors or shields is that analog measurements are less reliable due to process variations, a problem further exacerbated by feature scaling.

10.3.2.3 t-Private Circuits

The t-private circuit technique is proposed in [103] based on the assumption that the number of concurrent probe channels that an attacker could use is limited, and exhausting this resource deters an attack. In this technique, the circuit of a security-critical block is transformed so that at least $t + 1$ probes are required within one clock cycle to extract one bit of information. First, masking is applied to split computation into multiple separate variables, where an important binary signal x is encoded into $t + 1$ binary signals by XORing it with t independently generated random signals ($r_{t+1} = x \oplus r_1 \oplus \cdots \oplus r_t$) as shown in Fig. 10.23. Then, computations on x are performed in its encoded form in the transformed circuit. x can be recovered (decoded) by computing $x = r_1 \oplus \cdots \oplus r_t \oplus r_{t+1}$. The major issue with t-private circuit is that the area overhead involved for the transformation is prohibitively expensive.

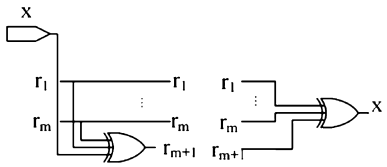

FIGURE 10.23

Input encoder (left) and output decoder (right) for masking in t-private circuits.

10.3.2.4 Other Countermeasure Designs

Some other countermeasures are implemented in real ICs, but less reported as novel designs because they are more or less dated. One known countermeasure that deters decapsulation stage of probing attacks is a light sensor that is sometimes included in a tamper-resistant design. Some other techniques include scrambling wires and avoiding repetitive patterns in shield mesh to impede the locating-target-wire stage of probing attacks. They are not particularly effective as exploits against them have been detailed in [97].

10.4 INVASIVE FAULT INJECTION ATTACK

Another type of physical attack that proved to be very effective to compromise cryptographic devices and processor's control flow is the invasive fault injection attack, which is realized through injecting faults by laser or focused ion beam (FIB) into a cryptographic device, and observing the corresponding outputs [104–106]. Using differential fault analysis (DFA) [107] methods, the secret key can be extracted. The fault injection techniques involved in this type of attack relies on having direct access to the silicon die, and the ability to target individual transistors in a very precise manner. These techniques are very powerful and have been demonstrated to be highly successful attack methods [108,109].

One example of optical fault injection techniques is a strong and precisely focused light beam that affects the behavior of one or more logic gates in a circuit. A strong radiation of a transistor may form a temporary conductive channel in the dielectric, which, in turn, may cause the switch of a state. For example, by targeting one of the transistors in a static random-access memory (SRAM) cell, the value stored in this cell could be flipped up or down at will [61,110].

A standard SRAM cell consists of six transistors, as shown in Fig. 10.24. Two pairs of p- and n-channel transistors create a flip-flop, while two other n-channel transistors are used for read and write. If the transistor VT1 could be opened for a very short time, then the state of the flip-flop could be changed. By exposing the transistor VT4 to light, the state of the cell would be switched to the opposite value. The main anticipated difficulties are: focusing the ionizing radiation down to several micrometers spot and choosing the proper intensity. The Microchip PIC16F84 microcontroller with 68 bytes of on-chip SRAM memory was used [110]. The light from a photo flash lamp was focused using the microscope optics. By shielding the light from the flash with an aperture made from aluminum foil, the state of only one cell can be changed. Focusing the light spot from the lamp on the area shown by

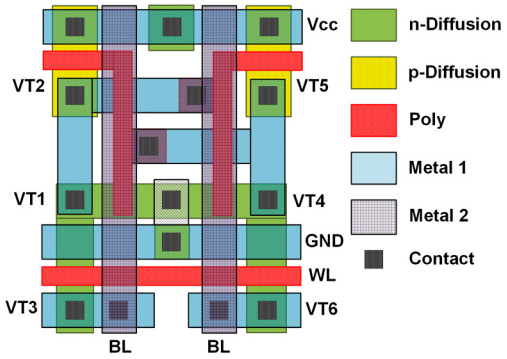

FIGURE 10.24

Layout of SRAM cell.

the white circle caused the cell to change its state from "1" to "0". By focusing the spot on the area shown by the black circle, the cell changed its state from "0" to "1", or remained in state "1".

Since the currents flowing inside a floating gate cell are much smaller than inside a SRAM cell, EPROM, EEPROM, and Flash memory cells are more vulnerable to fault injection attacks. EEPROM and Flash memory devices can be attacked by local heating technique [111], which use lasers to achieve modification. This was implemented with inexpensive laser diode module mounted on a microscope. The contents of the memory can be altered by locally heating up a memory cell inside a memory array, which can compromise the security of a semiconductor chip.

Nowadays, common optical fault injection facilities consist of a laser emitter, focusing lens, and a placement surface with stepper motors to achieve an accurate focusing of the beam. However, for this, or similar fault injection techniques, it is almost impossible to achieve subwavelength precision, which means the number of gates hit by the radiation is limited by the etching technology and the laser wavelength.

Focused ion beam (FIB) is one of the most accurate and powerful fault injection technique that enables an attacker to edit the circuit, reconstruct missing buses, cut existing wires, and mill through layers by depositing or removing material on the circuit die. For instance, Torrance and James [112] reported a successful reconstruction of an entire read bus of a memory containing a cryptographic key without damaging the contents of the memory. State-of-the-art FIBs can operate at a precision of 1 nm, that is, less than a tenth of the gate width of the smallest etchable transistor. FIB workstations require very expensive consumables, and a strong technical background to fully exploit their capabilities.

The countermeasures to prevent FIB-based fault injection attacks are almost the same with probing attacks as illustrated in Section 10.3.2. The basic strategies to prevent against fault injection attacks are intrusion detection, algorithmic resistance, and error detection. Common countermeasures against fault injection attacks have been illustrated in previous chapters. Most of these countermeasures can also be used to prevent against invasive fault injection attacks.

10.5 EXERCISES

10.5.1 TRUE/FALSE QUESTIONS

1. Optical microscopy can be used to produce transistor images in latest technology node.
2. If each metal layer has the same wire width and space, internal layer is better than top layer to build active shield to prevent bypass attack.
3. Different doping profiles of a transistor are not easily detectable via optical microscopy.
4. Flash memory can be reverse engineered using x-ray technology.
5. If there is no specific mechanism to protect a chip against probing attacks, it is recommended to hide sensitive nets on lower metal layers, such as Metal 1 or Metal 2, to achieve more coverage from higher metal layers.

10.5.2 SHORT-ANSWER TYPE QUESTIONS

1. What are the differences between reverse engineering with honest and dishonest motivations?
2. List three categories of reverse engineering and their differences.
3. Identifying components on a PCB is an important step in PCB-level RE. However, would only reading the label and marks on the package be sufficient to identify the real component?
4. If KEY bits are the only asset in an encryption module and these KEY bits have been properly protected against probing attack, would it be fair to assume that this crypto hardware is probing-resistant design? Explain.
5. Assume that an asset wire is on Metal 2, and a shield is planned to be built on either Metal 7 or Metal 8 to prevent probing attack. In your opinion, which layer is better to use? Explain why. Also assume that a cone-shaped hole will be milled during the probing attack, and the metal on layers 7 and 8 has the same width and space. Hint: Only consider the geometric relation of asset and shield wires.
6. Illustrate the basic steps to perform a front-side electrical probing attack.
7. Compared to clock glitch-based fault injection attack, what are the pros and cons of a laser-based optical fault injection attack?
8. Can an attacker utilize modern optical or electron microscopy to reverse engineering EEPROM?

10.5.3 MATHEMATICAL PROBLEMS

1. Considering Fig. 10.25, at least how many images are needed to figure out the interconnections of this chip?
2. Assuming that a shield is placed on the top layer of the chip, the shield width is 150 nm, the shield space is 500 nm, the shield wire thickness is 200 nm, a target wire is on metal 2 and the shield to target layer depth is 5000 nm, what is the maximum FIB aspect ratio that this shield can protect against? [Hint: Considering that a complete cut of shield wires can be detected and only consider perpendicular milling.]
3. Assuming that a shield is placed horizontally on the top layer of the chip, the shield width is 150 nm, the shield wire thickness is 200 nm, a vertical target wire is on metal 2, the length of this target wire is 3000 nm, and the shield to target layer depth is 5000 nm, what is the maximum shield space to protect against FIB-based probing attack whose maximum aspect ratio is 6? How

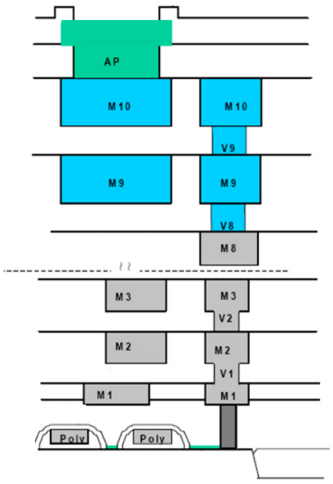

FIGURE 10.25

Interconnects of a chip.

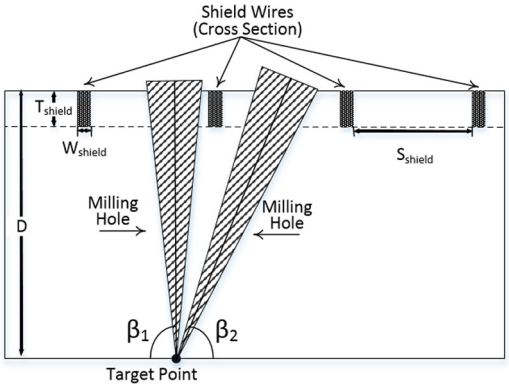

FIGURE 10.26

Probing attack scenarios.

many shield wires are needed to protect the target wire at least? Hint: Consider that a complete cut of shield wires can be detected and only consider perpendicular milling.

4. For Fig. 10.26, assume that a shield is placed vertically on the top layer of the chip, the shield width is 150 nm, the shield thickness is 200 nm, the shield spacing is 1 μm, a target probing point is located on M2 (in the quarter of two shield wires), the depth from shield layer to target point is 5 μm, a complete cut of shield wires can be detected, and a partial cut of shield wires is allowed. Answer the following questions:

(a) If only perpendicular milling is allowed ($\beta = 90°$), can a FIB with 5 aspect ratio probe the target point without a complete cut of any shield wire?

(b) If angled milling is allowed ($\beta \leq 90°$), can a FIB with 5 aspect ratio probe the target point without a complete cut of any shield wire?

REFERENCES

[1] I. McLoughlin, Secure embedded systems: the threat of reverse engineering, in: 2008 14th IEEE International Conference on Parallel and Distributed Systems, pp. 729–736.

[2] R.J. Abella, J.M. Daschbach, R.J. McNichols, Reverse engineering industrial applications, Computers and Industrial Engineering 26 (1994) 381–385.

[3] R. Torrance, D. James, The state-of-the-art in IC reverse engineering, in: C. Clavier, K. Gaj (Eds.), Cryptographic Hardware and Embedded Systems – CHES 2009, Springer Berlin Heidelberg, Berlin, Heidelberg, 2009, pp. 363–381.

[4] INSA, Interconnect design rules, Available at https://moodle.insa-toulouse.fr/pluginfile.php/2632/mod_resource/content/0/content/interconnect_design_rules.html.

[5] U. Guin, D. DiMase, M. Tehranipoor, Counterfeit integrated circuits: detection, avoidance, and the challenges ahead, Journal of Electronic Testing 30 (2014) 9–23.

[6] C. Bao, D. Forte, A. Srivastava, On application of one-class SVM to reverse engineering-based hardware Trojan detection, in: Fifteenth International Symposium on Quality Electronic Design, pp. 47–54.

[7] T.J. Biggerstaff, Design recovery for maintenance and reuse, Computer 22 (1989) 36–49.

[8] U. Guin, D. DiMase, M. Tehranipoor, A comprehensive framework for counterfeit defect coverage analysis and detection assessment, Journal of Electronic Testing 30 (2014) 25–40.

[9] S.K. Curtis, S.P. Harston, C.A. Mattson, The fundamentals of barriers to reverse engineering and their implementation into mechanical components, Research in Engineering Design 22 (2011) 245–261.

[10] M.T. Rahman, D. Forte, Q. Shi, G.K. Contreras, M. Tehranipoor, CSST: preventing distribution of unlicensed and rejected ICs by untrusted foundry and assembly, in: 2014 IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT), pp. 46–51.

[11] A. Baumgarten, A. Tyagi, J. Zambreno, Preventing IC piracy using reconfigurable logic barriers, IEEE Design Test of Computers 27 (2010) 66–75.

[12] SpacePhotonics, Anti-tamper technology, Available at http://www.spacephotonics.com/Anti_Tamper_Systems_Materials.php, 2013.

[13] DoD, Anti-tamper executive agent, Available at https://at.dod.mil/content/short-course, 2014.

[14] S.H. Weingart, Physical security devices for computer subsystems: a survey of attacks and defenses, in: Ç.K. Koç, C. Paar (Eds.), Cryptographic Hardware and Embedded Systems — CHES 2000, Springer Berlin Heidelberg, Berlin, Heidelberg, 2000, pp. 302–317.

[15] J.A. Roy, F. Koushanfar, I.L. Markov, EPIC: ending piracy of integrated circuits, in: 2008 Design, Automation and Test in Europe, pp. 1069–1074.

[16] Britannica.com, Integrated circuit (IC), Available at http://www.britannica.com/EBchecked/topic/289645/integrated-circuit-IC, 2014.

[17] MaximIntegrated, Glossary term: printed-circuit-board, Available at http://www.maximintegrated.com/en/glossary/definitions.mvp/term/Printed-Circuit-Board/gpk/973, 2014.

[18] Nikon, Microscopy, Available at http://www.microscopyu.com/, 2013.

[19] Nikon, Optical microscopy, Available at https://www.microscopyu.com/museum/model-smz1500-stereomicroscope.

[20] JEOL, Scanning electron microscope (SEM), Available at https://www.jeol.co.jp/en/products/list_sem.html.

[21] ZEISS, Transmission electron microscope (TEM), Available at http://jiam.utk.edu/facilities/microscopy/tem/index.php.

[22] Purdue.edu, Scanning electron microscope, Available at http://www.purdue.edu/ehps/rem/rs/sem.htm, 2014.

[23] SharedResources, Transmission electron microscope (TEM), Available at http://sharedresources.fhcrc.org/services/transmission-electron-microscopy-tem, 2014.

[24] Stanford.edu, Stanford microscopy facility, Available at https://microscopy.stanford.edu/, 2014.

[25] ThermoFisher, Focused ion beam (FIB), Available at https://www.fei.com/products/fib/.

[26] ZEISS, X-ray microscope, Available at https://www.zeiss.com/microscopy/int/products/x-ray-microscopy.html.
[27] FormFactor, Probe station, Available at https://www.formfactor.com/products/probe-systems/.
[28] GE, Inspection and NDT, Available at https://www.gemeasurement.com/inspection-and-nondestructive-testing, 2014.
[29] Tektronix, Logic analyzer, Available at https://www.tek.com/logic-analyzer.
[30] Grizzly, Computer numerical control (CNC), Available at http://users.dsic.upv.es/~jsilva/cnc/index.htm.
[31] R. Joshi, B.J. Shanker, Plastic chip carrier package, in: 1996 Proceedings 46th Electronic Components and Technology Conference, pp. 772–776.
[32] G. Phipps, Wire bond vs. flip chip packaging, Advanced Packaging Magazine 14, 7, 28, 2005.
[33] C. Tarnovsky, Deconstructing a 'secure' processor, in: Black hat federal, Available at http://www.blackhat.com/presentations/bh-dc-10/Tarnovsky_Chris/BlackHat-DC-2010-Tarnovsky-DASP-slides.pdf, 2010.
[34] SharedResources, Wet etching recipes, Available at http://www.eesemi.com/etch_recipes.htm, 2013.
[35] M.C. Hansen, H. Yalcin, J.P. Hayes, Unveiling the ISCAS-85 benchmarks: a case study in reverse engineering, IEEE Design Test of Computers 16 (1999) 72–80.
[36] W. Li, Z. Wasson, S.A. Seshia, Reverse engineering circuits using behavioral pattern mining, in: 2012 IEEE International Symposium on Hardware-Oriented Security and Trust, pp. 83–88.
[37] P. Subramanyan, N. Tsiskaridze, K. Pasricha, D. Reisman, A. Susnea, S. Malik, Reverse engineering digital circuits using functional analysis, in: 2013 Design, Automation Test in Europe Conference Exhibition (DATE), pp. 1277–1280.
[38] W. Li, A. Gascón, P. Subramanyan, W. Yang Tan, A. Tiwari, S. Malik, N. Shankar, S.A. Seshia, WordRev: finding word-level structures in a sea of bit-level gates, 2013, pp. 67–74.
[39] J. Rajendran, M. Sam, O. Sinanoglu, R. Karri, Security analysis of integrated circuit camouflaging, in: Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security, CCS '13, ACM, New York, NY, USA, 2013, pp. 709–720.
[40] SypherMedia, Circuit camouflage technology, Available at http://www.smi.tv/SMI_SypherMedia_Library_Intro.pdf, 2012.
[41] A.R. Desai, M.S. Hsiao, C. Wang, L. Nazhandali, S. Hall, Interlocking obfuscation for anti-tamper hardware, in: Proceedings of the Eighth Annual Cyber Security and Information Intelligence Research Workshop, CSIIRW '13, ACM, New York, NY, USA, 2013, 8.
[42] R.S. Chakraborty, S. Bhunia, Harpoon: an obfuscation-based SoC design methodology for hardware protection, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 28 (2009) 1493–1502.
[43] R. Maes, D. Schellekens, P. Tuyls, I. Verbauwhede, Analysis and design of active IC metering schemes, in: 2009 IEEE International Workshop on Hardware-Oriented Security and Trust, pp. 74–81.
[44] F. Koushanfar, Integrated circuits metering for piracy protection and digital rights management: an overview, in: Proceedings of the 21st Edition of the Great Lakes Symposium on Great Lakes Symposium on VLSI, GLSVLSI '11, ACM, New York, NY, USA, 2011, pp. 449–454.
[45] B. Gassend, D. Clarke, M. van Dijk, S. Devadas, Silicon physical random functions, in: Proceedings of the 9th ACM Conference on Computer and Communications Security, CCS '02, ACM, New York, NY, USA, 2002, pp. 148–160.
[46] J.A. Roy, F. Koushanfar, I.L. Markov, Protecting bus-based hardware IP by secret sharing, in: 2008 45th ACM/IEEE Design Automation Conference, pp. 846–851.
[47] G.K. Contreras, M.T. Rahman, M. Tehranipoor, Secure split-test for preventing IC piracy by untrusted foundry and assembly, in: 2013 IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFTS), pp. 196–203.
[48] J. Grand, Printed circuit board deconstruction techniques, in: 8th USENIX Workshop on Offensive Technologies (WOOT 14), USENIX Association, San Diego, CA, 2014.
[49] CTI, Counterfeit components avoidance program, Available at http://www.cti-us.com/CCAP.htm, 2013.
[50] DatasheetCatalog2013, Datasheet, Available at http://www.datasheetcatalog.com/, 2013.
[51] Alldatasheet, Electronic components datasheet search, Available at http://www.alldatasheet.com/, 2014.
[52] TechInsights, Sony Xperia play teardown and analysis, Available at http://www.techinsights.com/teardowns/sony-xperia-play-teardown/, 2014.
[53] X. Pan, Unified reconstruction theory for diffraction tomography, with consideration of noise control, Journal of the Optical Society of America A 15 (1998) 2312–2326.
[54] N. Asadizanjani, S. Shahbazmohamadi, E. Jordan, Investigation of Surface Geometry Thermal Barrier Coatings Using Computed X-Ray Tomography, vol. 35, 2015, pp. 175–187.

[55] S.E. Quadir, J. Chen, D. Forte, N. Asadizanjani, S. Shahbazmohamadi, L. Wang, J. Chandy, M. Tehranipoor, A survey on chip to system reverse engineering, ACM Journal on Emerging Technologies in Computing Systems 13 (2016) 6.
[56] B. Naveen, K.S. Raghunathan, An automatic netlist-to-schematic generator, IEEE Design Test of Computers 10 (1993) 36–41.
[57] W.-Y. Wu, M.-J.J. Wang, C.-M. Liu, Automated inspection of printed circuit boards through machine vision, Computers in Industry 28 (1996) 103–111.
[58] Ruzinoor Che Mat, Shahrul Azmi, Ruslizam Daud, Abdul Nasir Zulkifli, Farzana Kabir Ahmad, Morphological operation on printed circuit board (PCB) reverse engineering using MATLAB, Proc. Knowl. Manage. Int. Conf. Exhibit. (KMICE) (2006) 529–533.
[59] C. Koutsougeras, N. Bourbakis, V. Gallardo, Reverse engineering of real PCB level design using VERILOG HDL, International Journal of Engineering Intelligent Systems for Electrical Engineering and Communications 10 (2) (2002) 63–68.
[60] H.G. Longbotham, P. Yan, H.N. Kothari, J. Zhou, Nondestructive reverse engineering of trace maps in multilayered PCBs, in: AUTOTESTCON '95. Systems Readiness: Test Technology for the 21st Century. Conference Record, pp. 390–397.
[61] S.P. Skorobogatov, Semi-invasive attacks – a new approach to hardware security analysis, Technical Report UCAM–CL–TR–630, University of Cambridge Computer Laboratory, April 2005.
[62] F. Beck, Integrated Circuit Failure Analysis: A Guide to Preparation Techniques, John Wiley & Sons, 1998.
[63] C.D. Nardi, R. Desplats, P. Perdu, F. Beaudoin, J.-L. Gauffier, Oxide charge measurements in EEPROM devices, in: Proceedings of the 16th European Symposium on Reliability of Electron Devices, Failure Physics and Analysis, Microelectronics Reliability 45 (2005) 1514–1519.
[64] C. DeNardi, R. Desplats, P. Perdu, J.-L. Gauffier, C. Guérin, Descrambling and data reading techniques for flash-EEPROM memories. Application to smart cards, in: Proceedings of the 17th European Symposium on Reliability of Electron Devices, Failure Physics and Analysis. Wuppertal, Germany 3rd–6th October 2006, Microelectronics Reliability 46 (2006) 1569–1574.
[65] C. De Nardi, R. Desplats, P. Perdu, C. Guérin, J. Luc Gauffier, T.B. Amundsen, Direct measurements of charge in floating gate transistor channels of flash memories using scanning capacitance microscopy 2006, 2006.
[66] NREL, Scanning Kelvin probe microscopy, Available at http://www.nrel.gov/pv/measurements/scanning_kelvin.html, 2014.
[67] B. Bhushan, H. Fuchs, M. Tomitori, Applied Scanning Probe Methods X: Biomimetics and Industrial Applications, vol. 9, Springer, 2008.
[68] T. Wollinger, J. Guajardo, C. Paar, Security on FPGAs: state-of-the-art implementations and attacks, ACM Transactions on Embedded Computing Systems (TECS) 3 (2004) 534–574.
[69] A. Moradi, A. Barenghi, T. Kasper, C. Paar, On the vulnerability of FPGA bitstream encryption against power analysis attacks: extracting keys from Xilinx Virtex-II FPGAs, in: Proceedings of the 18th ACM Conference on Computer and Communications Security, CCS '11, ACM, New York, NY, USA, 2011, pp. 111–124.
[70] A. Moradi, M. Kasper, C. Paar, Black-box side-channel attacks highlight the importance of countermeasures: an analysis of the Xilinx Virtex-4 and Virtex-5 bitstream encryption mechanism, in: Proceedings of the 12th Conference on Topics in Cryptology, CT-RSA'12, Springer-Verlag, Berlin, Heidelberg, 2012, pp. 1–18.
[71] P. Swierczynski, A. Moradi, D. Oswald, C. Paar, Physical security evaluation of the bitstream encryption mechanism of Altera Stratix II and Stratix III FPGAs, ACM Transactions on Reconfigurable Technology and Systems 7 (2014) 34.
[72] S. Drimer, Volatile FPGA design security – a survey, in: IEEE Computer Society Annual Volume, IEEE, Los Alamitos, CA, 2008, pp. 292–297.
[73] E. Brier, C. Clavier, F. Olivier, Correlation power analysis with a leakage model, in: M. Joye, J.-J. Quisquater (Eds.), Cryptographic Hardware and Embedded Systems – CHES 2004, Springer Berlin Heidelberg, Berlin, Heidelberg, 2004, pp. 16–29.
[74] S. Drimer, Security for volatile FPGAs, Technical report UCAM-CL-TR-763, University of Cambridge, Computer Laboratory, 2009.
[75] D. Ziener, S. Assmus, J. Teich, Identifying FPGA IP-cores based on lookup table content analysis, in: 2006 International Conference on Field Programmable Logic and Applications, pp. 1–6.
[76] J.-B. Note, E. Rannaud, From the bitstream to the netlist, in: Proceedings of the 16th International ACM/SIGDA Symposium on Field Programmable Gate Arrays, FPGA '08, ACM, New York, NY, USA, 2008, pp. 264–271.
[77] F. Benz, A. Seffrin, S.A. Huss, Bil: a tool-chain for bitstream reverse-engineering, in: 22nd International Conference on Field Programmable Logic and Applications (FPL), pp. 735–738.

[78] B. Vajana, M. Patelmo, Mask programmed ROM inviolable by reverse engineering inspections and method of fabrication, 2002, US Patent App. 10/056,564.
[79] L. Chow, W. Clark, G. Harbison, J. Baukus, Use of silicon block process step to camouflage a false transistor, 2007, US Patent App. 11/208,470.
[80] H. Mio, F. Kreupl, IC chip with nanowires, 2008, US Patent 7,339,186.
[81] H.K. Cha, I. Yun, J. Kim, B.C. So, K. Chun, I. Nam, K. Lee, A 32-KB standard CMOS antifuse one-time programmable ROM embedded in a 16-bit microcontroller, IEEE Journal of Solid-State Circuits 41 (2006) 2115–2124.
[82] B. Stamme, Anti-fuse memory provides robust, secure NVM option, Available at http://www.eetimes.com/document.asp?doc_id=1279746, 2014.
[83] J. Lipman, Why replacing ROM with 1T-OTP makes sense, Available at http://www.chipestimate.com/tech-talks/2008/03/11/Sidense-Why-Replacing-ROM-with-1T-OTP-Makes-Sense, 2014.
[84] VirageLogic, Design security in nonvolatile Flash and antifuse FPGAs (NVM), Available at http://www.flashmemorysummit.com/English/Collaterals/Proceedings/2009/20090811_F1A_Zajac.pdf, 2014.
[85] G. Bartley, T. Christensen, P. Dahlen, E. John, Implementing tamper evident and resistant detection through modulation of capacitance, 2010, US Patent App. 12/359,484.
[86] D.H. Kim, K. Athikulwongse, S.K. Lim, A study of through-silicon-via impact on the 3d stacked IC layout, in: 2009 IEEE/ACM International Conference on Computer-Aided Design – Digest of Technical Papers, pp. 674–680.
[87] J. Van Geloven, P. Tuyls, R. Wolters, N. Verhaegh, Tamper-resistant semiconductor device and methods of manufacturing thereof, 2012, US Patent 8,143,705.
[88] F. Zachariasse, Semiconductor device with backside tamper protection, 2012, US Patent 8,198,641.
[89] S. Skorobogatov, Data remanence in flash memory devices, in: J.R. Rao, B. Sunar (Eds.), Cryptographic Hardware and Embedded Systems – CHES 2005, Springer Berlin Heidelberg, Berlin, Heidelberg, 2005, pp. 339–353.
[90] P. Thanigai, Introducing advanced security to low-power applications with FRAM-based MCUs, Available at http://www.ecnmag.com/articles/2014/03/introducing-advancedsecurity-low-power-applications-fram-mcus, 2014.
[91] Actel, Design security in nonvolatile Flash and antifuse FPGAs, Available at http://www.actel.com/documents/DesignSecurity_WP.pdf, 2002.
[92] K. Tiri, I. Verbauwhede, A dynamic and differential CMOS logic style to resist power and timing attacks on security IC's, Cryptology ePrint Archive, Report 2004/066, 2004.
[93] J. Wu, Y.-B. Kim, M. Choi, Low-power side-channel attack-resistant asynchronous S-box design for AES cryptosystems, in: Proceedings of the 20th Symposium on Great Lakes Symposium on VLSI, GLSVLSI '10, ACM, New York, NY, USA, 2010, pp. 459–464.
[94] L. Benini, E. Omerbegovic, A. Macii, M. Poncino, E. Macii, F. Pro, Energy-aware design techniques for differential power analysis protection, in: Proceedings 2003. Design Automation Conference (IEEE Cat. No. 03CH37451), pp. 36–41.
[95] H. Wang, D. Forte, M.M. Tehranipoor, Q. Shi, Probing attacks on integrated circuits: challenges and research opportunities, IEEE Design Test 34 (2017) 63–71.
[96] S. Skorobogatov, The bumpy road toward iPhone 5c NAND mirroring, ArXiv preprint arXiv:1609.04327, 2016, Available at https://arxiv.org/ftp/arxiv/papers/1609/1609.04327.pdf.
[97] C. Tarnovsky, Security failures in secure devices, in: Proc. Black Hat DC Presentation, 74, Feb. 2008, Available at http://www.blackhat.com/presentations/bh-dc-08/Tarnovsky/Presentation/bh-dc-08-tarnovsky.pdf, 2008.
[98] ARMInc., Building a secure system using TrustZone technology, Available at http://infocenter.arm.com/help/topic/com.arm.doc.prd29-genc-009492c/PRD29-GENC-009492C_trustzone_security_whitepaper.pdf, 2017.
[99] C. Boit, C. Helfmeier, U. Kerst, Security risks posed by modern IC debug and diagnosis tools, in: 2013 Workshop on Fault Diagnosis and Tolerance in Cryptography, pp. 3–11.
[100] S. Skorobogatov, Physical attacks on tamper resistance: progress and lessons, in: Proc. 2nd ARO Special Workshop Hardware Assurance, Washington, DC, USA, 2011, Available at http://www.cl.cam.ac.uk/sps32/ARO_2011.pdf.
[101] J.M. Cioranesco, J.L. Danger, T. Graba, S. Guilley, Y. Mathieu, D. Naccache, X.T. Ngo, Cryptographically secure shields, in: 2014 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST), pp. 25–31.
[102] S. Manich, M.S. Wamser, G. Sigl, Detection of probing attempts in secure ICs, in: 2012 IEEE International Symposium on Hardware-Oriented Security and Trust, pp. 134–139.
[103] Y. Ishai, A. Sahai, D. Wagner, Private circuits: securing hardware against probing attacks, in: D. Boneh (Ed.), Advances in Cryptology – CRYPTO 2003, Springer Berlin Heidelberg, Berlin, Heidelberg, 2003, pp. 463–481.
[104] A. Barenghi, L. Breveglieri, I. Koren, D. Naccache, Fault injection attacks on cryptographic devices: theory, practice, and countermeasures, Proceedings of the IEEE 100 (2012) 3056–3076.

[105] R. Anderson, M. Kuhn, Low cost attacks on tamper resistant devices, in: B. Christianson, B. Crispo, M. Lomas, M. Roe (Eds.), Security Protocols, Springer Berlin Heidelberg, Berlin, Heidelberg, 1998, pp. 125–136.
[106] D. Boneh, R.A. DeMillo, R.J. Lipton, On the importance of eliminating errors in cryptographic computations, Journal of Cryptology 14 (2001) 101–119.
[107] E. Biham, A. Shamir, Differential fault analysis of secret key cryptosystems, in: B.S. Kaliski (Ed.), Advances in Cryptology — CRYPTO '97, Springer Berlin Heidelberg, Berlin, Heidelberg, 1997, pp. 513–525.
[108] D. Boneh, R.A. DeMillo, R.J. Lipton, On the importance of checking cryptographic protocols for faults, in: W. Fumy (Ed.), Advances in Cryptology — EUROCRYPT '97, Springer Berlin Heidelberg, Berlin, Heidelberg, 1997, pp. 37–51.
[109] F. Bao, R.H. Deng, Y. Han, A. Jeng, A.D. Narasimhalu, T. Ngair, Breaking public key cryptosystems on tamper resistant devices in the presence of transient faults, in: B. Christianson, B. Crispo, M. Lomas, M. Roe (Eds.), Security Protocols, Springer Berlin Heidelberg, Berlin, Heidelberg, 1998, pp. 115–124.
[110] S.P. Skorobogatov, R.J. Anderson, Optical fault induction attacks, in: B.S. Kaliski, ç.K. Koç, C. Paar (Eds.), Cryptographic Hardware and Embedded Systems – CHES 2002, Springer Berlin Heidelberg, Berlin, Heidelberg, 2003, pp. 2–12.
[111] S. Skorobogatov, Local heating attacks on flash memory devices, in: 2009 IEEE International Workshop on Hardware-Oriented Security and Trust, July 2009, pp. 1–6.
[112] R. Torrance, D. James, The state-of-the-art in IC reverse engineering, in: Proceedings of the 11th International Workshop on Cryptographic Hardware and Embedded Systems, CHES '09, Springer-Verlag, Berlin, Heidelberg, 2009, pp. 363–381.