A Survey on Chip to System Reverse Engineering

SHAHED E. QUADIR, University of Connecticut
JUNLIN CHEN, University of Connecticut
DOMENIC FORTE, University of Connecticut
NAVID ASADIZANJANI, University of Connecticut
SINA SHAHBAZMOHAMADI, University of Connecticut
LEI WANG, University of Connecticut
JOHN CHANDY, University of Connecticut
MARK TEHRANIPOOR, University of Connecticut

The reverse engineering (RE) of electronic chips and systems can be used with honest and dishonest intentions. To inhibit reverse engineering (RE) for those with dishonest intentions (e.g., piracy and counterfeiting), it is important that the community is aware of the state-of-the-art capabilities available to attackers today. In this paper, we will be presenting a survey of reverse engineering and anti-reverse engineering techniques on the chip, board, and system levels. We also highlight the current challenges and limitations of anti-RE and the research needed to overcome them. This survey should be of interest to both governmental and industrial bodies whose critical systems and intellectual property (IP) require protection from foreign enemies and counterfeiters who possess advanced RE capabilities.

Categories and Subject Descriptors: B.7.1 [**Integrated Circuits**]: Types and Design Styles

General Terms: Economics, Human factors, Legal aspects, Performance, Reliability, Security

Additional Key Words and Phrases: Reverse engineering (RE), anti-reverse engineering (anti-RE), hardware security, delayering, firmware extraction, FPGA security

ACM Reference Format:
Shahed E. Quadir, Junlin Chen, Domenic Forte, Navid Asadizanjani, Sina Shahbazmohamadi, Lei Wang, John Chandy and Mark Tehranipoor, 2015. A Survey on Chip to System Reverse Engineering. *ACM J. Emerg. Technol. Comput. Syst.* V, N, Article acmArticle (Month 2015), 33 pages.
DOI: http://dx.doi.org/10.1145/0000000.0000000

1. INTRODUCTION

Reverse engineering (RE) is the process by which an object is examined in order to gain a full understanding of its construction and/or functionality. RE is now widely used to disassemble systems and devices in a number of different contexts, such as industrial design, cloning, duplication, and reproduction [McLoughlin 2008]. In this paper, we will be focusing on the reverse engineering of electronic systems, which can be achieved by extracting their underlying physical information using destructive and nondestructive methods [Abella et al. 1994] [Torrance and James 2009].

The motivation for RE could be "honest" or "dishonest" as shown in Table I [Abt and Pawlowicz 2012] [Guin et al. 2014b] [Bao et al. 2014]. Those with "honest" intentions

Author's addresses: S. E. Quadir, J. Chen, D. Forte, N. Asadizanjani, L. Wang, J. Chandy and M. Tehranipoor, Electrical and Computer Engineering Department, University of Connecticut; S. Shahbazmohamadi (Current address) Mechanical Engineering Department, Manhattan College, Riverdale, NY 10471.

DOI: http://dx.doi.org/10.1145/0000000.0000000

tend to perform RE for the following reasons: verification, fault analysis, research and development, and education about the workings of an existing product. In many countries, RE is legal as long as patents and design copyrights are not violated [Biggerstaff 1989]. When RE is performed to clone, pirate or counterfeit a design, to develop an attack, or insert a hardware Trojan, these are considered "dishonest" intentions. If the functionality of a cloned system is close enough to the original, for example, then the "dishonest" entity or individuals could sell large amounts of counterfeit products without the prohibitive research and development costs required by the owner of the original [Guin et al. 2014a].

Table I. The Motivation for Reverse Engineering (RE)

"Honest" Intentions	**"Dishonest" Intentions**
Failure analysis and defect identification	Fault injection attacks
Detection of counterfeit products [Guin et al. 2014b] [Guin et al. 2014]	Counterfeiting
Circuit analysis to recover manufacturing defects	Tampering
Confirmation of intellectual property (IP)	IP piracy and theft
Hardware Trojan detection [Bao et al. 2014]	Hardware Trojan insertion
Analysis of a competitor's product, Obsolete product analysis	Illegal cloning of a product
Education and research	Development of attacks

There are several examples of reverse engineering (RE) and cloning of systems throughout history. During World War II, an American B-29 bomber was captured, reverse-engineered, and copied by the former Soviet Union [Curtis et al. 2011]. The original and the clone (Tupolev Tu-4 bomber) are shown in Figure 1. The only difference between the B-29 and Tu-4 are the engines and cannons. Another example of RE took place during the Vietnam War [Radovich and Worms 2014]. In the late 1960s, the AQM-34G-R model of the United States Unmanned Aerial Vehicles (UAV), also known as a "drone" was lost in the mainland of China. The American technology was analyzed to create a replica called the WuZhen-5. The WuZhen-5 was subsequently used in China's invasion of Vietnam in 1979. Even today, the US Department of Defense (DoD) is concerned about RE attempts being made against U.S. weapons systems [DoD 2014].

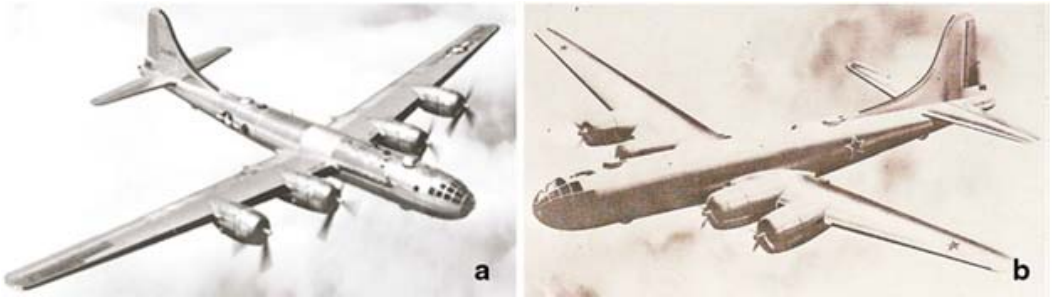


Fig. 1. An example of reverse engineering (RE) from World War II; (a) United States Air Force B-29 bomber and (b) Soviet Union Tupolev Tu-4 bomber which is a reverse-engineered copy of the B-29 [Curtis et al. 2011].

Aside from RE of large systems, secret information such as critical design and personal information can also be extracted or cloned from electronic chips and printed circuit boards (PCBs). For example, the simple structure and increasing reliance on commercial-off-the-shelf components makes it very easy to RE and clone a PCB. Reverse engineering of PCB and ICs could also result in the development of future attacks against them. For example, many smartcards today contain ICs that store personal information and perform transactions [Radovich and Worms 2014]. "Dishonest" parties

could reverse engineer these ICs to access the confidential information of the card holder, commit financial crimes, etc.

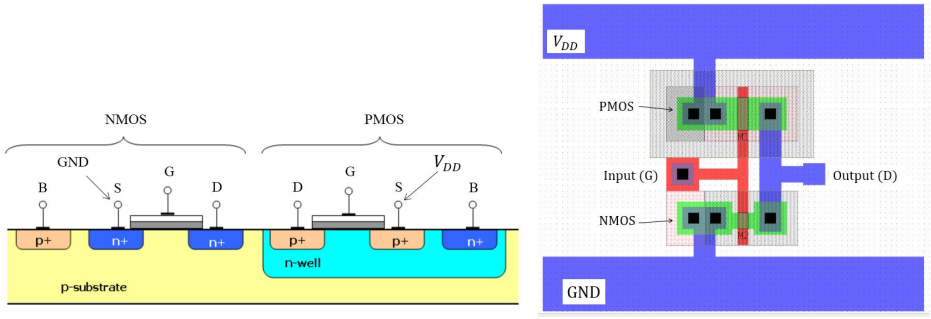
Another concern in the electronics industry is IC piracy using RE [Radovich and Worms 2014] [Rahman et al. 2014]. In 2010, Semiconductor Equipment and Materials International (SEMI) published a survey about intellectual property (IP) infringement. The survey revealed that, of the 90% of companies that have experienced IP infringement, 54% faced serious infringement of their products [Baumgarten et al. 2010]. Many "dishonest" companies can illegally copy the circuit and technology in order to mass-produce and sell pirated copies in the open market without authorization. On a related note, counterfeiting of ICs through RE is also a concern for military and industrials sectors. Counterfeit electronics also result in unrecoverable losses for the IP owner. Counterfeit ICs and systems may be tampered or otherwise less reliable, resulting in vulnerabilities and life-threatening issues.

To summarize, reverse engineering (RE) is a longstanding problem that is of great concern to today's governments, militaries, and various industries due to the following: (1) the attacks and security breaches that could occur through the RE of classified military systems, financial systems, etc.; (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, which can result from the counterfeiting of products through the use of RE; (4) the negative impact that RE has on new product innovations, incentives for research and development, and - by extension - the worldwide job market.

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 United States DoD is currently conducting research on anti-RE technologies that may prevent classified data, weapons, and IP from being compromised by foreign adversaries [Photonics 2013]. The objective of the DoD's anti-tamper program is to obstruct unapproved technology transfer, maximize the costs of RE, enhance U.S./coalition military capacities, train the DoD community, and educate the DoD community on anti-tampering technologies [DoD 2014]. Unfortunately, most of this work 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 techniques. For example, tamper resistant materials and sensors have been used to resist theft or reverse engineering (RE) [Weingart 2000]. 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 pico-probing attempts, single chip coatings have also been applied. Many different packaging techniques could be used to protect a device: brittle packages, aluminum packages, polished packages, bleeding paint, as well as holographic and other tamper responding tapes and labels [Weingart 2000]. 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 insulating have been used to block X-ray imaging attempts.

Also, obfuscation software and hardware security primitives have been used for the protection of systems and software. These anti-RE techniques would 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 protection, and tamper responding technology [Weingart 2000] [Roy et al. 2008a].



(a) (b)

Fig. 2. (a) Simplified cross sectional view [Wikipedia.org 2010] and (b) layout of a CMOS inverter [Swarthmore.edu 2005].

In this survey, we shall cover the reverse engineering (RE) of electronic devices from chip to system levels:

1) ***Chip-level reverse engineering (RE):*** A chip is an integrated circuit 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 [Wikipedia.org 2010] [Britannica.com 2014]. In Figure 2(a) and Figure 2(b), simplified cross sectional view and layout of a CMOS inverter are shown respectively. In Figure 2(a), 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 V_{DD} line, and the source (S) of the NMOS is connected to a metal ground (GND) line. The drains (D) of the PMOS 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 [Radovich and Worms 2014]. X-ray tomography is a nondestructive method of RE, which 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 reverse engineering (RE):*** Electronic chips and components are mounted on a laminated non-conductive printed circuit board (PCB) [Integrated 2014] and electrically interconnected using conductive copper traces and vias [Wikipedia.org 2014]. The board might be single- or multi-layered depending on the complexity of the electronic system. Reverse engineering of PCBs begins with the identification of the components mounted on the board, its traces on the top and bottom (visible) layers, its ports, etc. 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 reverse engineering (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 non-volatile memories (NVMs), such as ROM, EEPROM, and Flash. For more advanced designs with

FPGAs (such as Xilinx FPGAs), the firmware-like netlists are also stored within the NVM memories. By reading out and analyzing the contents in the memory, reverse engineering can provide a deeper insight into the system under attack.

Based on the discussions above, the taxonomy of reverse engineering (RE) is shown in Figure 3. First, reverse engineering is performed to tear down the product or system in order to identify the sub-systems, packages, and other components. The sub-systems could be electrical or mechanical. In this paper, we will focus on electrical sub-systems. The electrical sub-systems under analysis consist of hardware and firmware. A reverse engineer could analyze the FPGA, board, chip, memory, and software to extract all information. This paper is concerned with RE when it is done with bad intentions and with anti-RE as a remedy against this form of RE. We examine this type of RE and anti-RE for each level, including equipment, techniques, and materials.

The rest of the paper is organized as follows: In the next section, we will introduce the imaging and other specialized equipment that can be used for RE. In Sections III and IV, we shall focus on RE and anti-RE at the chip-level. We will discuss board-level RE and anti-RE techniques in Section V and Section VI, respectively. Section VII will present system-level RE, and Section VIII will discuss anti-RE at the system-level. Challenges and directions for future research are presented in Section IX. Finally, we will conclude the paper in Section X.

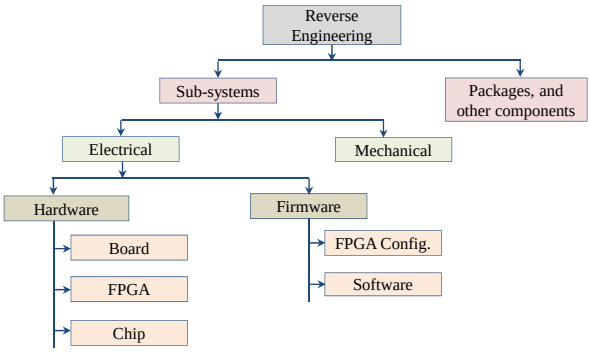


Fig. 3. Taxonomy of reverse engineering (RE).

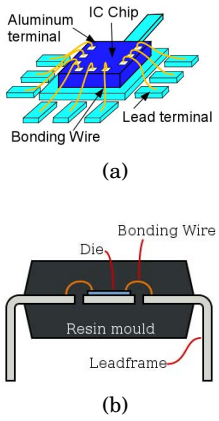


Fig. 4. IC parts (a) top view [Group 2011] and (b) cross sectional view [Answers.com 2014].

2. EQUIPMENT

Advanced RE requires different kinds of specialized equipment. Throughout the paper, we will refer to this equipment. Therefore, a short summary of each is provided below:

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 [Nikon 2013]. 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, the use of optical microscopes can only be used to analyze PCB and chip exteriors because the resolution is too low for current chip feature sizes (<100 nm).

Scanning Electron Microscopes (SEM): In a scanning electron microscope (SEM), focused beams of electrons are used to produce images [Purdue.edu 2014]. For a sam-

ple, the electrons interact with atoms, a process that produces signals for detection. Reverse engineers should start with a cross-section of an unknown chip. The scanning electron microscopes (SEM) could be used for analyzing the cross-section, as well as the composition and thickness of each layer of the die. The object could be magnified by 10 times to approximately 30,000 times. The scanning electron microscope (SEM) provides the following advantages over traditional microscopes:

— ***Higher resolution:*** The SEM has higher resolution and, with high magnification, it can resolve the features on the sub-micron level.
— ***Large depth of field:*** When a specimen (such as the internal elements of a chip) is focused 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.

Transmission Electron Microscopes (TEM): With transmission electron microscopes (TEM), a beam of electrons is transmitted through and interacts with a sample [Sharedresources 2014]. Like SEMs, transmission electron microscopes (TEM) have a very high spatial resolution, which can provide detailed information about the internal structures of a sample [Stanford.edu 2014]. Also, TEM can be used to view a chip's cross-section and its internal layers.

Focused Ion Beam (FIB): The working principle of a focused ion beam (FIB) is the same as a scanning electron microscope (SEM) except that, instead of using an electron beam, an ion beam is used. The ion beam enables one to do material deposition and removal with nanometer resolution, which can be used for TEM sample preparation, circuit editing, etc. 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 (SCM): 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 [Torrance and James 2009]. A probe electrode is applied at the top of the sample surface, and this electrode then scans across the sample. The change in electrostatic capacitance between the surface and the probe is used for obtaining information about the sample [GE 2014b].

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 [GE 2014a].

Probe Stations: Probe stations are highly-precise manual probe units for wafers and substrates. They support a wide variety of electrical measuring, device and wafer characterization (DWC), failure analysis (FA), 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, while 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 reverse engineering (RE) at the chip, board, and system levels. In the case of FPGA bitstream reverse engineering (RE), the logic analyzer can

be adopted to measure the JTAG communication signals between FPGA and external memory.

Computer Numerical Control (CNC): 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. CNCs can run mills, lathes, grinds, plasma cutters, laser cuts, etc. The motion is controlled along all three main axes, which enables three dimensional process.

3. CHIP-LEVEL REVERSE ENGINEERING (RE)

An integrated circuit (IC) typically consists of a die, a lead frame, wire bonding, and molding encapsulant [Yener 2014] as shown in Figure 4.

The package of a chip can be classified in different ways. The materials that are used can be ceramic or plastic [Joshi and Shanker 1996]. As ceramics are costly, plastics are commonly used as the package material. Packaging can also be wire-bond or flip-chip [Phipps 2005]. 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 an IC 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 [Phipps 2005].

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 subsections below.

3.1. Decapsulation

First, reverse engineers identify the package materials and remove the chip's packaging. Depot is the traditional method by which acid solution is used for removing the package [Torrance and James 2009]. 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 [Torrance and James 2009].

To remove the die package, one can use selective or non-selective methods. Wet chemical etching and plasma etching can be used as selective techniques, while non-selective techniques would be thermal shock, grinding, cutting, and laser ablation

[Yener 2014]. Different kinds of decapsulation methods - along with their pros and cons - are shown in Table II.

Table II. 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 high etch rate Works well when die size is small compared with package	Does not work with ceramic packages Acid can damage lead frame and bond wires Isotropic etch
	Dry	Remove 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 backside of the die Does not work on specific area
	Milling	Remove material in the 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 to damage die Not controllable for a specific area
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 $\mu m^3/s$)
	Plasma FIB	High accuracy in material removal (nm) Can be performed on controlled area Faster milling rate (2000 $\mu m^3/s$)	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

After decapsulation, the die needs to be cleaned before delayering and/or imaging can be performed because dust may be present, resulting in artifacts [Tarnovsky 2010]. Different methods for cleaning the dust are outlined below [Yener 2014]:

— ***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 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.

3.2. 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. Figure 5 shows the cross-section of a CMOS chip with three metal layers. The knowledge from cross-sectional imaging is critical as it determines how the

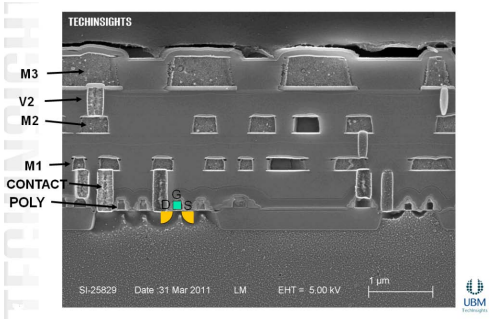


Fig. 5. SEM cross section image of metal layers of a chip for CMOS technology [Abt and Pawlowicz 2012].

Table III. Wet Etching Recipes for Different Types of Metals and Etching Process [Siliconfareast 2013]

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 microns/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 microns/min @ 25 C
Silicon nitride (Si_3N_4)	Refluxing phosphoric acid	-	Use at 180C; 6.5 nm/min @ 25 C; Plasma etching is preferred for removing Si_3N_4

delayering must be performed (i.e., how thick are the layers, what types of conductors are used, etc.).

Several methods can be used simultaneously when a chip is delayered - methods such as wet/plasma etching, grinding, and polishing. A reverse engineer should determine the etchants needed and the time needed to remove each layer because the layout could be dependent 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 III [Siliconfareast 2013].

Once the etchants are determined for delayering a specific layer and metal, a reverse engineer will begin with etching the passivation layer; then the reverse engineer will take an image of the highest metal layer; and, 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 [Torrance and James 2009] [Abt and Pawlowicz 2012]. Also, 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 [Abt and Pawlowicz 2012]: a semi-automated polishing machine, a semi-automated milling machine, a laser, a gel etch, a computer numerical control (CNC) milling machine, and an ion beam milling machine.

When the chip has been delayered, one could face the following challenges [Abt and Pawlowicz 2012]:

— ***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. But, 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 (i.e., 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 (e.g., removing a metal layer without affecting the vias).

3.3. Imaging

During the delayering process, thousands of high-resolution images are taken to capture all 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 2.

3.4. Post-Processing

The post-processing or circuit extraction after delayering consists of the following steps: (i) image processing, (ii) annotation, (iii) gate-level schematic extraction, (iv) schematic analysis and organization, and (v) high-level netlist extraction from the gate-level schematic. Each of these steps is described in greater detail below.

3.4.1. Image Processing. Taking images manually is becoming increasingly difficult because the size of the ICs is shrinking, along with many of their features [Torrance and James 2009]. Advanced electrical labs now use automated instruments (X-rays, SEMs, digital microscopes), which 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. Also it is important to establish the lineup of the layers' contacts and vias before the extraction.

3.4.2. 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 the imaged layers of the chip and align them for extraction [Torrance and James 2009]. 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 facilitate the extraction of large blocks of digital cells quickly.

3.4.3. Gate-Level Schematic Extraction. Sometimes the images are imperfect, as the images may be taken manually. Also 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 [Torrance and James 2009]. 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 are dependent on each other, so changing one could affect the others.

3.4.4. 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 facilitate an analysis of the architecture and circuit design. Some structures, such as differential pairs and bandgap references, could be easily recognizable.

3.4.5. High-Level Netlist Extraction from Gate-Level Schematic. After circuit extraction is performed on the stripped IC (derivation of circuit schematic diagram), several techniques [Hansen et al. 1999] [Li et al. 2012] [Li et al. 2013] [Subramanyan et al. 2013] could be applied to get the high-level description for analysis and validation of the functionality of the chip using simulation. [Hansen et al. 1999] propose reverse engineering (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. [Li et al. 2012] present 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. The authors in [Li et al. 2013] propose an automatic way to derive word-level structures which 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 is used in [Subramanyan et al. 2013] 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.

4. CHIP-LEVEL ANTI-REVERSE ENGINEERING

There are several approaches for the anti-reverse engineering of integrated circuits, which include camouflage, obfuscation, and other techniques. These methods are described in more detail below.

4.1. Camouflage

Layout level techniques such as cell camouflage [Rajendran et al. 2013] [SypherMedia 2012] 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 Figure 6. In Figure 6(a) and Figure 6(b), the layouts of two-input NAND and NOR gates are shown. These gates functionalities can be easily identified by their layouts. In contrast, Figure 6(c) and Figure 6(d) 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 Figure 6(a) and Figure 6(b)). Camouflaging (see Figure 6(c) and Figure 6(d)) 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.

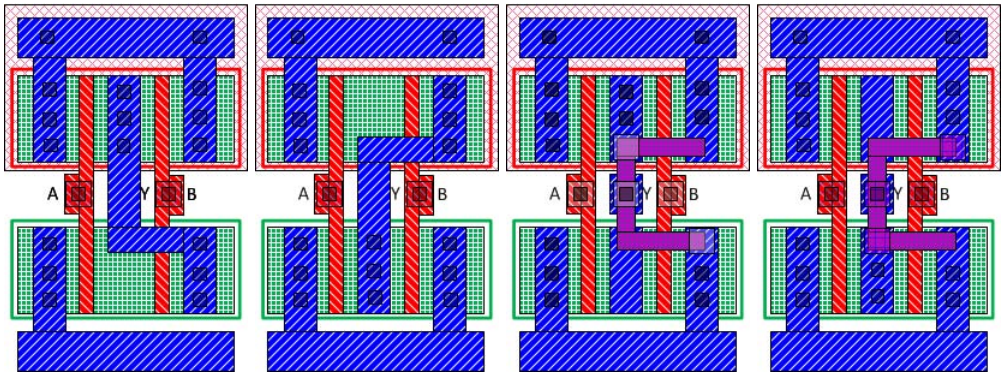


Fig. 6. Standard (a) NAND gate and (b) NOR gate. These gates could be easily differentiable by looking at the top metal layers. Camouflaged (c) NAND gate and (d) NOR gate. These gates have identical top metal layers and are, therefore, harder to identify [Rajendran et al. 2013].

4.2. Obfuscation

Obfuscation techniques entail making a design or system more complicated in order to prevent reverse engineering (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 [Desai et al. 2013] [Chakraborty and Bhunia 2009]. The HARPOON (HARdware Protection through Obfuscation Of Netlist) method could be used against piracy and tampering, and the technique could provide protection at every level of the hardware design and manufacturing process [Chakraborty and Bhunia 2009]. The proposed 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 obfuscated mode to reach normal mode only for specific input vectors, which are known as the “key” for the circuit.

[Desai et al. 2013] 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 (obfuscated) and functional mode. The functional mode will be 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 reverse engineer by increasing the interaction with the state machine. Furthermore, 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 proposed 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.

4.3. Other Techniques

Today, most companies are fabless, meaning that the fabrication of chips is outsourced. A semiconductor foundry is given the design [Maes et al. 2009] 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 [Koushanfar 2011] [Rahman et al. 2014]. 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 as secret keys to protect from cloning [Gassend et al. 2002] [Koushanfar 2011]. PUF is very 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.

The authors in [Baumgarten et al. 2010] have proposed a reconfigurable logic barrier scheme which 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 4.2 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 EPIC (End Piracy of Integrated Circuits) [Roy et al. 2008a]. This key is produced by the IP holder and is 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 reverse engineering (RE) and controls how many chips should be made. The EPIC technique's limitations include complex communication to the IP holder, which could impact test time and time to market. Also, this technique requires higher levels of power consumption.

[Roy et al. 2008b] 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.

5. BOARD-LEVEL REVERSE ENGINEERING (RE)

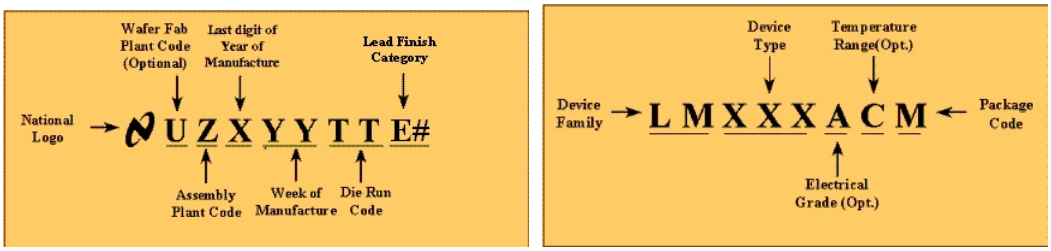
The goal of board-level RE is to identify all components on the board and the connections between them. All of the components used in a design are called the bill of materials (BOM) [McLoughlin 2008]. The components and parts of a printed circuit board (PCB) could be any of the following: microprocessors, 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 [Torrance and James 2009] [Grand 2014]. To identify the components, test points, and parts of the PCB, silkscreen markings are often used [McLoughlin 2008]. 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 semi-custom 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 [McLoughlin 2008] because a custom device could be undocumented, or documentation could be provided only under a non-disclosure agreement.

IC markings can be divided into the following four parts [CTI 2013]:

— 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. And, 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 Figure 7. 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 [DatasheetCatalog 2013] [Alldatasheet 2014].



(a) (b)

Fig. 7. Marking convention on the Texas Instruments (TI) chips (a) first line and (b) second line [Instruments 2014].

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 [CTI 2013]. The die marking could help identify the mask number, part number, date of the die mask completion or copyright registration, company logo, and the trademark symbol. An example of the die marking on a Texas Instruments (TI) 65 nm baseband processor is shown in Figure 8. A die marking could match the 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 [Techinsights 2014], so if someone can find the functionality of one chip, then they 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 [Techinsights 2014]. The only difference between these two chips is their clock speed.

Fig. 8. Die marking on a Texas Instruments (TI) 65 nm processor [Quirk 2013].

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, as well as 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 the outer layers' components are captured [McLoughlin 2008]. 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 [Grand 2014]. 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 [McLoughlin 2008].

Non-destructive 3D Imaging of PCBs Using X-ray Tomography: X-ray tomography is a non-invasive imaging technique that makes it possible to visualize the internal structure of an object without the interference of over- and under-layer structures. The principle of this method is to acquire a stack of two dimensional (2D) images and then use mathematical algorithms such as the direct Fourier transform and center slice theory [Pan 1998] to reconstruct the three dimensional (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, are important to consider in the selection of the tomography process parameters: source/detector distance to object, source power, detector objective, filter, exposure time, number of projections, center shift, and beam hardening. Internal and external structures will be ready to analyze when the 3D image is reconstructed [Grand 2014]. A discussion of how to select the right values for any of these parameters is outside the scope of this paper. More information on tomography parameters is available in [Asadizanjani et al.].

As an example, we have analyzed the traces and via holes of a four-layer custom PCB using a Zeiss Versa 510 X-ray machine. To make sure that we can observe features on the board, we 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 and can be performed without the need for oversight, and it should be widely applicable to most PCBs.

For the four-layer custom board in Figure 9, all traces, connections, and via holes are clearly captured. In order to validate the effectiveness of the tomography approach, the results are compared with the board design files previously used to produce the PCB. The board includes a frontside, backside, 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 Figure 10.

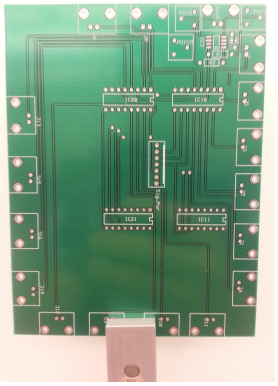


Fig. 9. PCB mounted in sample holder.

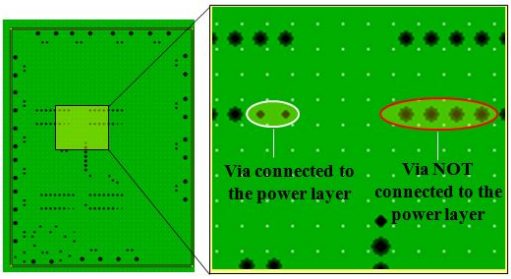


Fig. 10. Layout design of the internal power layer.

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 microns). In Figure 11 one slice is provided, which shows the information of the internal power layer.

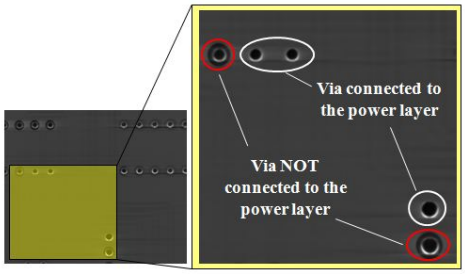
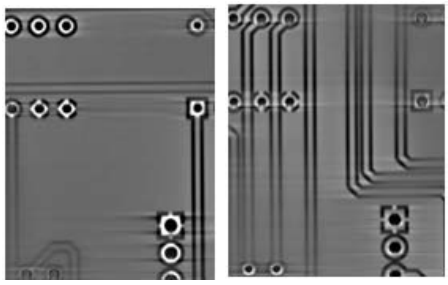


Fig. 11. Virtual slicing presents power layer.



(a) (b)

Fig. 12. Reconstructed (a) top and (b) bottom layers.

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. So, one can easily determine which via holes are connected to an internal layer. The same prin-

ciple will let us detect the traces on the side layers of the board due to the attendance of copper on the traces, as shown in Figure 12.

Netlist Extraction After Imaging: After capturing images of the PCB via delayering or X-ray tomography, connections between all 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 [Naveen and Raghunathan 1993]. 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 [Wu et al. 1996] [Mat et al. 2006] [Koutsougeras et al. 2002] [Longbotham et al. 1995] [Johnson 2013]. In [Wu et al. 1996], a visual inspection system is used 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. An image of the raw PCB is read in [Mat et al. 2006] and then a structuring element is applied to an input image 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. 2002] 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. In [Longbotham et al. 1995], the layers of the Circuit Card Assemblies/ Printed Circuit Boards (CCA/PCB) are separated using X-ray stereo imaging. The focus is to identify the solder joints and traces on the different layers of a multi-layered PCB. In the automated process technique [Johnson 2013], photos are taken from one- or two-layer printed circuit boards (PCBs). Then, a C++ program is used to automatically reverse engineer the netlist.

6. PCB-LEVEL ANTI-REVERSE ENGINEERING

Ensuring the complete protection from PCB-level RE is a difficult task, 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 is provided below [McLoughlin 2008]:

(1) Tamper-proof fittings (such as torx), custom screws 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. Also, 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 board. 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 (for instance, wiring connections between unused pins to unused

pins, having spare inputs and outputs from processors to route signals, dynamically jumbling buses, and jumbling the PCB silkscreen annotations) could complicate the RE process. However, such techniques also require the use of more complex chip and complicated design methods.

Many of the above methods are difficult to implement and could significantly increase design and manufacturing costs. Table IV shows the effectiveness of anti-RE techniques at the board-level [McLoughlin 2008]. A total of five levels are used for scaling based on identifying design cost, manufacturing impact, and reverse engineering (RE) cost.

Table IV. Implementation Challenges of Anti-RE Techniques for Board-Level, where Very High = Most and Very Low = Least

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 (ball grid array)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

7. SYSTEM-LEVEL REVERSE ENGINEERING (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. We refer to this as system-level RE.

Parallel to the embedded system design involving ASICs and MCU/DSPs 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 3). In this section, FPGA RE is categorized into the system-level RE, as well, since both the firmware in MCUs, DSPs, etc. and netlist information are stored in the nonvolatile memory (NVM) devices. Note that we primarily focus on the SRAM-based FPGAs in this section due to its largest market share among the reconfigurable hardware devices.

In this section, we will first introduce the storing in these NVM devices, and then describe the RE methods used to extract the firmware/netlist accordingly.

7.1. 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 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 [Skorobogatov 2005b] as shown in Figure 13.

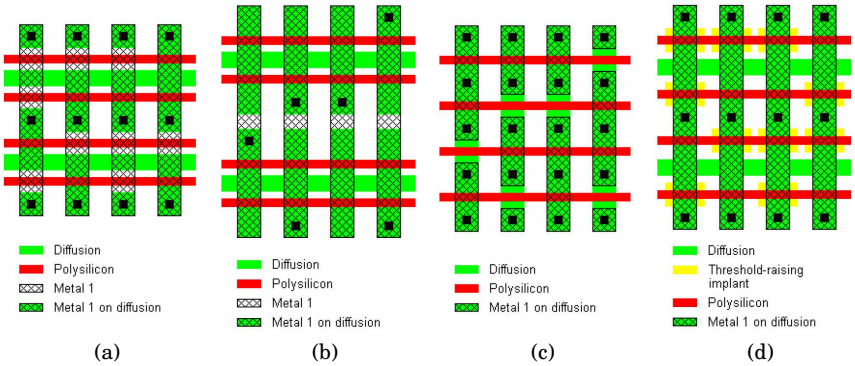


Fig. 13. Illustrations of (a) active layer programming ROM, (b) contact layer programming ROM, (c) metal layer programming ROM, and (d) implant programming ROM [Skorobogatov 2005b].

— ***Active layer programming ROM***: The logic state is represented by the presence or absence of a transistor. As shown in Figure 13(a), 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 Figure 13(b).
— ***Metal layer programming ROM***: The binary information is encoded by short-circuiting the transistor or not as shown in Figure 13(c).
— ***Implant programming ROM***: The different logic state is achieved by different doping levels in the diffusion area (see Figure 13(d)). Generally, higher doping levels will raise the on/off voltage threshold, which will disable the transistor.

Compared with ROM, EEPROM provides the users with the capability to reprogram the contents. As shown in Figure 14(a), one bit cell of EEPROM is composed of two transistors—floating gate transistor (FGT) and select transistor (ST). The floating gate transistor is feathered with two stacked gates: a control gate (CG) and a floating gate (FG). The logic state of the bitcell 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 Figure 14(b)) has almost the same structure as EEPROM except for the absence of ST, which is irrelevant to the logic state and just 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. As shown in Figure 15, the logic blocks will be 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 the information about the netlist can be obtained from the bitstream file.

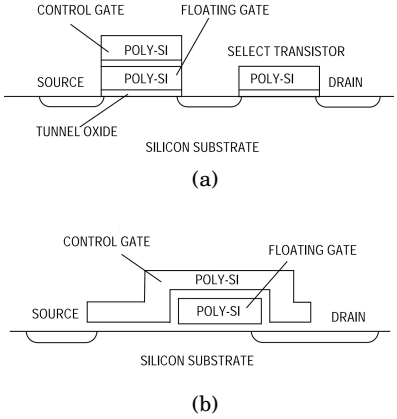


Fig. 14. Illustrations of (a) EEPROM and (b) Flash [Ledford 2004].

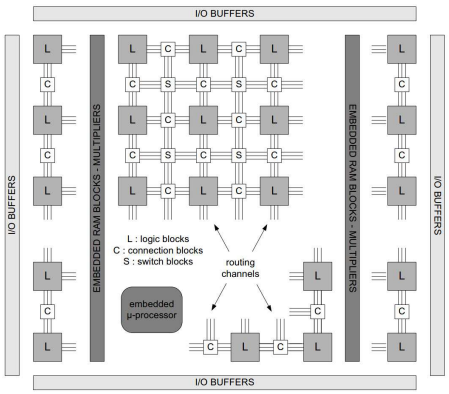


Fig. 15. FPGA hardware block diagram [Standaert 2008].

7.2. ROM Reverse Engineering (RE)

To reverse engineer the ROM contents, one can take advantage of modern optical and electron microscopy to observe the binary states of each cell.

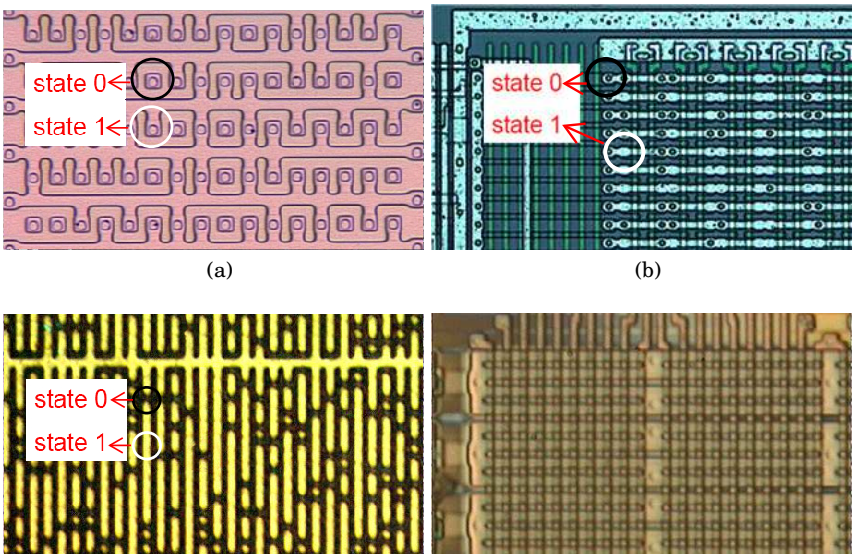
— ***Active layer programming ROM***: The metal layer and poly layer need to be removed using the delayering approaches discussed in Section 3, for they will obscure the active layer underneath. In Figure 16(a), the two different states can be visible.
— ***Contact layer programming ROM***: It is much easier to reverse engineer this kind of ROM since 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. The presence and absence of contacts are shown in Figure 16(b).
— ***Metal layer programming ROM***: This type of ROM can be directly observed under a microscope without having to perform any delayering process, as shown in Figure 16(c).
— ***Implant programming ROM***: This type of ROM is inherently resistant to optical microscopy since different logic states appear identical as in Figure 16(d). To observe the impact of different doping levels, additional dopant-selective crystallographic etch techniques [Beck 1998] should be utilized to separate the two logic states as in Figure 17.

Generally, ROM only provides limited protection against reverse engineering (RE). Among all types of ROM, the metal layer programming ROM offers the worst security due to the fact that the metal layer is easy to obtain with little effort, while the implant programming ROM provides the highest level of protection available.

7.3. EEPROM/Flash Reverse Engineering (RE)

Since EEPROM and Flash memory have similar structures and the same logic storage mechanism (as discussed above), they often can be reverse engineered by the same procedures. Due to the fact that EEPROM/Flash represents different states by the electrons - not by the geometric difference, X-Ray technology cannot be used to detect the contents. Further, any attempt to delayer and measure the electrons in the floating gate, such as SEM and TEM, will change the electron distribution, thereby disturbing the contents inside.

For a quite long time, the EEPROM/Flash technology has been regarded as the most robust memory defense against RE. Recently, several methods [De Nardi et al.



(a) (b)

(c) (d)

Fig. 16. Optical inspection of (a) active layer programming rom [Kömmerling and Kuhn 1999], (b) contact layer programming ROM [Yener 2014], (c) metal layer programming ROM [Brightsight 2014], and (d) implant programming ROM before selective etch [Brightsight 2014]

2005] [De Nardi et al. 2006b] [De Nardi et al. 2006a], though very expensive and requiring specialized equipment, were proposed to extract the contents in EEPROM/Flash correctly. Note that both the below methods occur from the backside of the memory, since traditional frontside delaying and imaging will cause the charges in the floating gate (FG) to vanish [De Nardi et al. 2005].

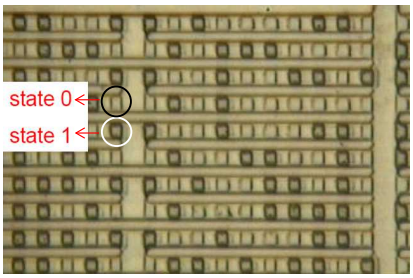
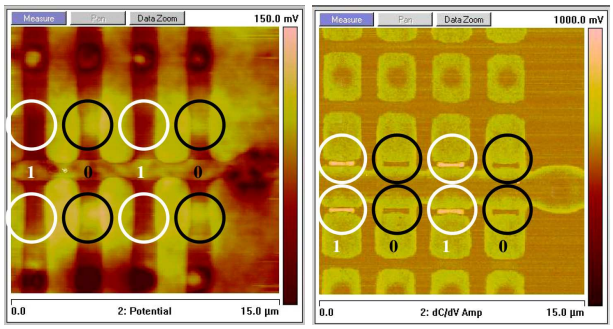


Fig. 17. Optical inspection of implant programming ROM after selective etch [Brightsight 2014].



(a) (b)

Fig. 18. (a) SKPM scan and (b) SCM scan from the backside of Flash memory [De Nardi et al. 2006a].

7.3.1. Scanning Kelvin Probe Microscopy (SKPM) Procedure. The SKPM procedure [NREL 2014] directly probes the floating gate (FG) potential through the tunnel oxide layer with a thickness of 10nm, which isolates the FG with the transistor channel as illustrated in Figure 14(a). So the first step is to remove the silicon from the backside 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 Figure 18(a), the scanning data from SKPM shows the two-dimensional 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 Figure 18(a)).

7.3.2. Scanning Capacitance Microscopy (SCM) Procedure. Unlike the SKPM procedure, the SCM procedure will measure the capacitance variations between the tip with the sample in the contact mode and the high-sensitivity capacitance sensor equipped on the SCM [Bhushan et al. 2008]. 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 backside delayering should keep a silicon thickness of 50-300 nm to leave the transistor channel undamaged. Then the bit information can be read as depicted in Figure 18(b). 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 V. Note that, with technology scaling, the electrons stored in the FG have been reduced to fewer than 1000 electrons for 90nm-node NAND Flash [De Nardi et al. 2006a]. In this case, the SKPM procedure can no longer recognize two logic states accurately, while the SCM still performs well.

Table V. Comparison between SKPM and SCM Procedures

Property	SKPM Procedure	SCM Procedure
Delayering position	Backside	Backside
Delayering depth	Entire silicon	50−300 nm thickness
Sensitivity	Low	High
Measured Carriers	Electrons	Holes
Measured parameter	Potential	Capacitance
Operation mode	Non-contact	Contact
Application	All EEPROM and some Flash	All EEPROM and Flash

7.4. Reverse Engineering (RE) of FPGAs

FPGA reverse engineering (RE) involves analyzing the configuration bitstream file and transforming the bitstream file into the hardware netlist, which consists of all the components and interconnections at the register transfer level (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.

7.4.1. 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 wire-tap the JTAG data and command lines to capture the communication between the FPGA and Flash memory during startup.

7.4.2. Bitstream Decryption. To increase the security level of FPGA, most FPGA manufacturers will encrypt 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) [Wollinger et al. 2004]. Now the wire-tapped encrypted

bitstreams will not yield any information for reverse engineering (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, since they will trigger tamper detection in the FPGA to zeroize the secret keys. So far, no public report exists on a successful invasive attack towards SRAM-based FPGA.

Recently, it has been reported that the bitstream encryption of several mainstream FPGA series [Moradi et al. 2011] [Moradi et al. 2012] [Swierczynski et al. 2013] is vulnerable to the side-channel attacks [Drimer 2008]. Basically, a side-channel attack (SCA) is a non-invasive attack to exploit the relationship between physical information (power, timing, and electromagnetic emanation) and certain hardware operations in the FPGA implementation. In [Moradi et al. 2011], the triple DES encrypted bitstream file from Xilinx Virtex-II Pro FPGA has been first successfully cracked by the side-channel attack. 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 in order to guess and verify a small portion of the key (e.g., 6-bit 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 [Moradi et al. 2012] by a more sophisticated type of correlation power analysis [Brier et al. 2004].

In a similar way, the FPGA power consumption or electro-magnetic radiation (EM) 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 side-channel attack [Swierczynski et al. 2013]. The fact that all the above attacks can be conducted within several hours reveals the vulnerability of the bitstream encryption.

7.4.3. Bitstream Reversal. Prior to converting the bitstream file into the corresponding hardware netlist, one should first understand the bitstream structure which is usually documented by FPGA vendors and is accessible online. Typically, a bitstream file consists of four parts [Drimer 2009]: command header, configuration payload, command footer, and start-up sequence. In the case of Xilinx FPGA, the configuration payload determines the configuration points (LUT or Lookup Table, memory, register, multiplexer, etc.) 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 last 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 LUT (Lookup Table), CLB (Configurable Logic Block), and multiplier from the bitstream file. [Ziener et al. 2006] shows the possibility to identify the embedded IP-cores by extracting the contents of LUT in Xilinx Virtex-II FPGA.

Full bitstream reversal: [Note and Rannaud 2008] makes the first public attempt to convert the bitstream file into the netlist. The set-theoretic algorithm and cross-correlation algorithm [Note and Rannaud 2008] were used to build a database linking the bitstream bits to the associated resources (configuration points and programmable

interconnect points) in 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 because it only relies on the information from the accessible XDL file (Xilinx Design Language) 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 [Benz et al. 2012], 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 [Benz et al. 2012] indicate new issues that the cross-correlation algorithm cannot perfectly relate all the resources in FPGA with the bits in bitstream file. Therefore, the immature technique of the bitstream reversal makes the FPGA embedded system more robust against FPGA reverse engineering (RE) compared with ASIC design MCU designs.

8. SYSTEM-LEVEL ANTI-REVERSE ENGINEERING

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

8.1. Anti-reverse Engineering 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 will make all the memory cells identical under optical inspection, no matter what the contents. This type of solution, though it 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 7.1, the use of different doping levels to encode information constitutes one kind of camouflage technique. Several other camouflage techniques are provided below.

8.1.1. Camouflage Contacts. Different from the contact layer programming ROM (see Figure 13(b)), 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 [Patelmo and Vajana 2001]. To decode the contents, careful chemical etching has to be applied to find the real contacts, and this is very time consuming. From the viewpoint of time/cost, this technique will also increase production periods and lower the manufacturing yield.

8.1.2. Camouflage Transistors. To improve the security of active layer programming ROM (see Figure 13(a)), false transistors are made to confuse the RE attempts, instead of using the absence of transistors [Baukus et al. 2005]. The false transistors, essentially with no electrical functions, have the same top-down view as the true transistors under 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, while it only requires minimal effort during manufacturing.

8.1.3. 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 [Mio and Kreupl 2008]. The real connections between bit lines and word lines act as transistors, while the non-electrical 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 biggest challenge with camouflage nanowires, however, is to manufacture the ROM at a high enough volume and a high enough yield, given the restrictions of our current technology.

Practically, all the above 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.

8.1.4. 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, while they make the design and fabrication process much more complicated. Currently, ROM replacements (such as 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 [Cha et al. 2006] [Stamme 2014] [Lipman 2014], the split channel 1T transistor anti fuse [Lipman 2014] exhibits many advantages over the conventional ROM with respect to cell area, access speed, and immunity to RE. As shown in Figure 19(a), the anti-fuse 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 anti-fuse, existing RE techniques (such as delayering from either frontside or backside, FIB based voltage contrast [Logic 2014], and top-down view or cross-sectional view from electrical microscopy) won't expose any information contained, not to mention the fact that it is very difficult to locate the oxide breakdown. Additionally, the anti-fuse memory is compatible with the standard CMOS technology, thus no additional masks or processing steps are required for fabrication. Considering the security, performance, and cost, the anti-fuse memory may eventually replace current ROM devices with the feature size continuously scaling down [Stamme 2014].

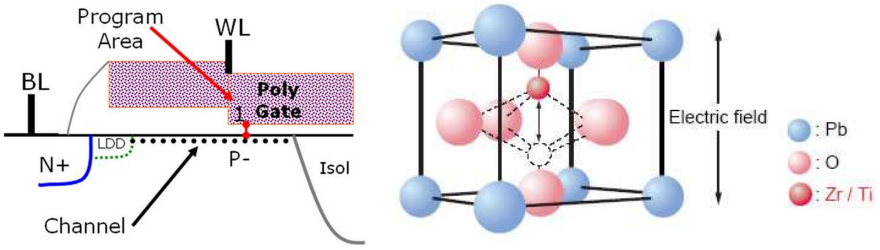


Fig. 19. Illustrations of one memory cell (a) anti fuse-OTP [Lipman 2014] and (b) FeRAM [Fujitsu 2014]

8.2. Anti-reverse Engineering for EEPROMs/Flashs

To reverse engineer the EEPROM/Flash memory, attackers prefer to delayer from the backside to avoid disturbing the floating charges. Thus, the most effective countermeasure would be to prevent backside attacks. Here we will first briefly introduce some backside attack detection methods, and then we will review one alternative to EEPROM/Flash, which can inherently tolerate the backside attacks.

8.2.1. Circuit Parameter Sensing. Performing the delayering process from the backside will thin the bulk silicon. By burying two parallel plates in the bulk silicon to form a capacitor, the capacitance sensing [Bartley et al. 2011] will detect the capacitance reduction when the attacker polishes from the backside. 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 (TSV) technique [Kim et al. 2009], makes it much easier to fabricate. Similarly, other parameters, such as resistance [Van Geloven et al. 2012], can be measured and compared with the pre-defined reference resistance threshold.

8.2.2. Light Sensing. By optically monitoring the backside 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 [Zachariasse 2012]. 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 for the sensing methods introduced in Sections 8.2.1 and 8.2.2, once the detection signal generated from the above 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 Focused Ion Beam (FIB) to eventually render all detection-erasure methods useless. In addition, even if the memory successfully erases all the contents, the attackers still have the chance to determine the actual values according to the residual electrons on the floating gate due to data remanence [Skorobogatov 2005a].

8.2.3. FeRAM Memory. As previously mentioned, the use of electrons on floating gates to represent the logic states makes the EEPROM/Flash memory vulnerable to reverse engineering (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 as well as lower write power consumption. Recently, it was reported that FeRAM can still possess very strong protections for the contained state [Thanigai 2014].

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)O3) compound. As shown in Figure 19(b), 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 Figure 19(b)) is in the scale of nanometer, thereby exposing nothing to the top-down view. One possible attack to reveal the contents, though 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.

8.3. Anti-reverse Engineering for FPGAs

The fact that the encrypted SRAM FPGA can provide enough reverse engineering (RE) resilience leaves less space for the research and development of anti-RE techniques compared with the ASIC design. Nevertheless, we still categorize the existing FPGA anti-RE techniques into three groups according to the FPGA RE procedure.

8.3.1. Bitstream Hiding. By integrating the bitstream storage memory with FPGA, the Flash FPGA and antifuse FPGA [Actel 2002] do not require external configuration memory, leaving the direct wire-tapping useless. Unlike the SRAM FPGA, the Flash FPGA does not need bitstream download during power-up due to the Flash memory nonvolatility. The antifuse FPGA has been widely used in military applications because of its higher RE resilience. As we have discussed in Sections 7.3 and 8.1, 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. Though 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.

8.3.2. Side-channel Resistance. The recent success of side-channel attacks on FPGA prove 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 resistance design is to remove the dependency between deciphering operations and power consumption. [Tiri and Verbauwhede 2004] presents a dynamic and differential CMOS logic implementation which has a constant power consumption and circuit delay irrespective of different circuit operations. [Wu et al. 2010] proposes to adopt the asynchronous logic design to obtain power consumption independent of computations and data. These methods, while effective against SCA, lead to much larger area and power consumptions compared with 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 non-deterministic, 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 [Benini et al. 2003], the power reduction technique is proposed to lower the power consumption overhead from noise generation.

8.3.3. Bitstream Anti-reversal. 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 in order to impede bitstream reversal under non-invasive attacks.

Currently, bitstream reversal strongly depends on the amount of publically available information (e.g., user guides) and undocumented information (e.g., files generated by EDA tools). FPGA vendors should take the possibility of reverse engineering (RE) attacks into account when releasing new information in order 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, while other non-critical parts are still loaded from the external memory. This partial configuration only leaves the wire-tapper partial information about the whole FPGA mapping information, thereby fundamentally eliminating the potential of bitstream reversal.

8.4. Summary of Anti-RE Techniques for System-Level

Table VI illustrates the cost and the associated yield loss of the system level anti-RE techniques discussed below. To assess the feasibility of the anti-RE techniques, we roughly classify the costs of RE/anti-RE 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, while 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 VI 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 VI 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, bitsteam hiding stands out as the best candidate.

Table VI. Costs of Anti-RE Techniques and RE for System-Level, where Very High = Most and Very Low = Least

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
	Antifuse one-time programming	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 anti-reversal	Low	High	–

9. CHALLENGES AND FUTURE WORK

It is very hard to defeat reverse engineers in their attacking attempts, but a complex and protective anti-RE system could be devised that would be so time-consuming, difficult, and expensive that it could deter most forms of RE. In the meantime, if the reverse engineering of adversaries is successful, the technology could be superseded by its next-generation version [DoD 2014]. There are several anti-RE technologies discussed in Sections 4, 6 and 8, but it is worth mentioning that some of these techniques are vulnerable to RE attacks. For example, camouflage tries to make imaging, etc. difficult. Therefore, if someone delayers the chip, board, or system destructively, they could find the full functionality. Obfuscation doesn't prevent imaging but makes the functionality of the design ambiguous or locked. After conducting a destructive analysis, a reverse engineer could extract high-level netlists so that they could find the functionality [Rajendran et al. 2012]. There are other, chip-level techniques like hardware metering, EPIC, and reconfigurable logic barriers that could be used as anti-RE techniques, but most of these methods are used for anti-piracy. Metering, reconfigurable logic barriers, and EPIC have limited uses for anti-RE because, if someone can extract the key by applying backside attacks, RE is trivial. Also, applying anti-RE techniques to the board-level can be very challenging because the board is much more vulnerable to RE due to its simple laminated structure.

Also, most of the existing anti-RE techniques are merely deployed against RE attacks from the front side of the system, leaving the system more vulnerable to the backside attacks [Boit et al. 2013]. As discussed in Section 7.3, it is much easier and quicker to decapsulate the chip from the backside to expose the memory contents. New

dedicated countermeasures against backside attacks are very much in demand in the future from the system design perspective. Though some of the anti-RE techniques used against EEPROMs/Flashs reverse engineering are deployed particularly for the backside attacks (see Section 8.2), most of them rely largely on the embedded power supply (such as embedded battery) to sense the invasion from the backside. From the attackers' perspective, these countermeasures can be broken without too much difficulty, since the active sensing mechanisms are easily bypassed once the power supply is recognized and isolated by the attackers.

Although a large amount of recent research and new developments have appeared on anti-RE techniques for decades, there still exist many open problems that will need additional research in the future. In the following section, we will list some important issues that still need to be addressed.

(1) Currently, measuring quantitatively how strong the anti-RE techniques are still remains an open challenge. Lack of metrics for evaluating the efficiency of anti-REs will delay their mass deployment in the industrial sector. By viewing side-channel attacks as a communication problem, a good measuring example in [Standaert 2008] [Standaert et al. 2006] demonstrates that both the leaked information obtained by side-channel attacks and the effect of the adopted countermeasures can be measured from the information theory viewpoint.
(2) In practice, anti-REs will inevitably cause other issues, such as reliability, power consumption and area overhead. The tradeoffs between reverse engineering resistance, reliability, power consumption, and area overhead should be thoroughly investigated before applying the anti-REs in different electrical systems.
(3) Currently, most anti-RE techniques are proposed independently. Integrating two or more anti-RE techniques in the same design can definitely improve the hardware security against reverse engineering. Take the active sensing mechanisms of anti-RE techniques for EEPROMs/Flash for example: the power supply can be hidden by the existing camouflage techniques to further increase the complexity of reverse engineering. Additionally, the cost of reverse engineering will be increased if BGA packages are used with multilayer PCBs [Skorobogatov 2005b]. In this way, the interconnections will be hard to observe, and BGA pins on chip will be impossible to access for analysis. It is worth mentioning that desoldering and the decapsulation of BGA packages are harder to break than plastic packages. Also, interconnection obfuscation could be applied by introducing dummy ICs in the PCBs [Ghosh et al. 2014]. This technique will scramble the traces of the board, so RE could not discover the exact design of the PCB. The problem of how to optimally combine different techniques, however, still remains an open issue.
(4) As far as we know, most of the current anti-RE techniques basically provide security features attached to the original designs, which do not consider anti-RE capability. In the long run, electrical systems are in urgent need of the research and development of new approaches with inherent resistance to reverse engineering. For example, as the technology of 3D IC matures, it is believed that the 3D structure will possess the inherent potential to resist reverse engineering because the die in 3D IC structures is less observable compared with traditional IC structures.
(5) To protect against noninvasive attacks, some dummy metals or ceramic powders could be used inside the internal structure of the chip or between the board layers without changing its functionality. These materials highly attenuate the X-ray and create artifacts in the reconstructed images of the tomography. Therefore, the reverse engineer cannot extract the desired information about the layout after X-ray imaging.

10. CONCLUSION

In this paper, we have presented reverse engineering (RE) techniques at the chip, board, and system levels, and anti-RE techniques to mitigate them. We have also offered a taxonomy of possible RE strategies. Anti-RE techniques are discussed elaborately in terms of their classification, costs, and effectiveness at each level. Since RE could lead to such serious problems as IP theft, piracy, and the cloning of products, this paper shall raise awareness of the current, state-of-the-art techniques and provide motivation for the development of new, low-cost, and robust anti-RE techniques at all levels. Finally, we provide future directions to improve the state-of-the-art in anti-RE at all three levels.

REFERENCES

Robert J. Abella, James M. Daschbach, and Roger J. McNichols. 1994. Reverse engineering industrial applications. *Computers & industrial engineering* 26, 2 (1994), 381–385.

Jason Abt and Chris Pawlowicz. 2012. Circuit analysis techniques: delayering and circuit vision. (2012). http://www.techinsights.com/.

Actel. 2002. Design Security in Nonvolatile Flash and Antifuse FPGAs. (2002). http://www.actel.com/documents/DesignSecurity_WP.pdf

Alldatasheet. 2014. Electronic Components Datasheet Search. (2014). http://www.alldatasheet.com/.

Answers.com. 2014. IC Construction. (2014). http://www.answers.com/topic/dual-in-line-package.

N. Asadizanjani et al. Investigation of surface geometry change in thermal barrier coatings using computed X-ray tomography. In *38th Int'l Conf and Expo on Advanced Ceramics and Composites, ICACC 2014*.

Chongxi Bao et al. 2014. On application of one-class SVM to reverse engineering-based hardware Trojan detection. In *Quality Electronic Design (ISQED), 2014 15th International Symposium on*. IEEE, 47–54.

G. K. Bartley, T. A. Christensen, P. E. Dahlen, and E. S. II John. 2011. Implementing tamper evident and resistant detection through modulation of capacitance. (Aug. 2 2011). US Patent 7,989,918.

James P. Baukus, Lap-Wai Chow, William M. Clark Jr, and Gavin J. Harbison. 2005. Use of silicon block process step to camouflage a false transistor. (Dec. 27 2005). US Patent 6,979,606.

Alex Baumgarten, Akhilesh Tyagi, and Joseph Zambreno. 2010. Preventing IC piracy using reconfigurable logic barriers. *IEEE Design and Test of Computers* 27, 1 (2010), 66–75.

Friedrich Beck. 1998. *Integrated circuit failure analysis: a guide to preparation techniques*. John Wiley & Sons.

Luca Benini et al. 2003. Energy-aware design techniques for differential power analysis protection. In *Design Automation Conference, 2003. Proceedings*. IEEE, 36–41.

F. Benz et al. 2012. Bil: A tool-chain for bitstream reverse-engineering. In *Field Programmable Logic and Applications (FPL), 2012 22nd International Conference on*. IEEE, 735–738.

Bharat Bhushan, Harald Fuchs, and Masahiko Tomitori. 2008. *Applied scanning probe methods X: biomimetics and industrial applications*. Vol. 9. Springer.

Ted J. Biggerstaff. 1989. Design recovery for maintenance and reuse. *Computer* 22, 7 (1989), 36–49.

Christian Boit, Clemens Helfmeier, and Uwe Kerst. 2013. Security Risks Posed by Modern IC Debug and Diagnosis Tools. In *Fault Diagnosis and Tolerance in Cryptography (FDTC), 2013 Workshop on*. IEEE, 3–11.

Eric Brier, Christophe Clavier, and Francis Olivier. 2004. Correlation power analysis with a leakage model. (2004), 16–29.

Brightsight. 2014. Physical Attacks on Cryptographic devices. (08 2014). http://www2.lirmm.fr/w3mic/SOCSIP/images2/stories/colloque2014/vanbattum.pdf.

Britannica.com. 2014. Integrated circuit IC. (2014). http://www.britannica.com/EBchecked/topic/289645/integrated-circuit-IC.

Hyouk-Kyu Cha, Ilhyun Yun, Jinbong Kim, Byeong-Cheol So, Kanghyup Chun, Ilku Nam, and Kwyro Lee. 2006. A 32-KB standard CMOS antifuse one-time programmable ROM embedded in a 16-bit microcontroller. *Solid-State Circuits, IEEE Journal of* 41, 9 (2006), 2115–2124.

Rajat S. Chakraborty and Swarup Bhunia. 2009. HARPOON: an obfuscation-based SoC design methodology for hardware protection. *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on* 28, 10 (2009), 1493–1502.

CTI. 2013. Counterfeit Components Avoidance Program. (2013). http://www.cti-us.com/CCAP.htm.

Shane K. Curtis et al. 2011. The fundamentals of barriers to reverse engineering and their implementation into mechanical components. *Research in Engineering Design* 22, 4 (2011), 245–261.

DatasheetCatalog. 2013. Datasheet. (2013). http://www.datasheetcatalog.com/.

Christophe De Nardi, Romain Desplats, Philippe Perdu, Felix Beaudoin, and J. L. Gauffier. 2005. Oxide charge measurements in EEPROM devices. *Microelectronics Reliability* 45, 9 (2005), 1514–1519.

Christophe De Nardi, Romain Desplats, Philippe Perdu, J. L. Gauffier, and C. Guérin. 2006a. Descrambling and data reading techniques for flash-EEPROM memories. Application to smart cards. *Microelectronics Reliability* 46, 9 (2006), 1569–1574.

Christophe De Nardi, Romain Desplats, Philippe Perdu, Christophe Guérin, Jean Luc Gauffier, and Thomas B. Amundsen. 2006b. Direct measurements of charge in floating gate transistor channels of flash memories using scanning capacitance microscopy. In *International Symposium for Testing and Failure Analysis*, Vol. 32. ASM International; 1998, 86.

Avinash R. Desai et al. 2013. Interlocking obfuscation for anti-tamper hardware. In *Proceedings of the Eighth Annual Cyber Security and Information Intelligence Research Workshop*. ACM, 8.

DoD. 2014. DoD Anti-Tamper. (2014). https://at.dod.mil/.

Saar Drimer. 2008. Volatile FPGA design security–a survey. *IEEE Computer Society Annual Volume* (2008), 292–297.

Saar Drimer. 2009. Security for volatile FPGAs. *Rapport technique UCAM-CLTR-763, University of Cambridge, Computer Laboratory* (2009).

Fujitsu. 2014. Ferroelectric RAM Technology. (08 2014). http://www.fujitsu.com/emea/services/microelectronics/fram/technology.html.

B. Gassend, D. Clarke, M. Van Dijk, and S. Devadas. 2002. Silicon physical random functions. In *Proceedings of the 9th ACM conference on Computer and communications security*. ACM, 148–160.

GE. 2014a. Radiography (X-ray)- Non-Destructive Testing. (2014). http://www.ge-mcs.com/en/radiography-x-ray.html.

GE. 2014b. Scanning Electron Microscope. (2014). http://www.ge-mcs.com/en/radiography-x-ray.html.

Swaroop Ghosh, Abhishek Basak, and Swarup Bhunia. 2014. How Secure Are Printed Circuit Boards Against Trojan Attacks? (08 2014).

Joe Grand. 2014. Printed circuit board deconstruction techniques. In *Proceedings of the 8th USENIX conference on Offensive Technologies*. USENIX Association, 11–11.

Tanaka Group. 2011. Bonding wire. (2011). http://www.tanaka-bondingwire.com/english/03-1bw.html.

Ujjwal Guin, Daniel DiMase, and M. Tehranipoor. 2014a. A comprehensive framework for counterfeit defect coverage analysis and detection assessment. *Journal of Electronic Testing* 30, 1 (2014), 25–40.

Ujjwal Guin, Daniel DiMase, and M. Tehranipoor. 2014b. Counterfeit integrated circuits: Detection, avoidance, and the challenges ahead. *Journal of Electronic Testing* 30, 1 (2014), 9–23.

Ujjwal Guin et al. 2014. Counterfeit Integrated Circuits: A Rising Threat in the Global Semiconductor Supply Chain. *Proc. IEEE* 102, 8 (2014), 1207–1228.

Mark C. Hansen, Hakan Yalcin, and John P. Hayes. 1999. Unveiling the ISCAS-85 benchmarks: A case study in reverse engineering. *IEEE Design & Test of Computers* 16, 3 (1999), 72–80.

Texas Instruments. 2014. Marking Convention. (2014). http://focus.ti.com/quality/docs/gencontent.tsp?templateId=5909&navigationId=12626&contentId=153966.

Maxim Integrated. 2014. Printed-Circuit-Board. (2014). http://www.maximintegrated.com/en/glossary/definitions.mvp/term/Printed-Circuit-Board/gpk/973.

Ben Johnson. 2013. EE368: Reverse Engineering of Printed Circuit Boards. (2013).

R. Joshi and B. J. Shanker. 1996. Plastic chip carrier package. In *Electronic Components and Technology Conference, 1996. Proceedings., 46th*. IEEE, 772–776.

Dae Hyun Kim, Krit Athikulwongse, and Sung Kyu Lim. 2009. A study of through-silicon-via impact on the 3D stacked IC layout. In *Proceedings of the 2009 International Conference on Computer-Aided Design*. ACM, 674–680.

Oliver Kömmerling and Markus G. Kuhn. 1999. Design principles for tamper-resistant smartcard processors. In *USENIX workshop on Smartcard Technology*, Vol. 12. 9–20.

Farinaz Koushanfar. 2011. 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*. ACM, 449–454.

C. Koutsougeras et al. 2002. 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.

Stephen Ledford. 2004. *Non-Volatile Memory Technology Overview*. Technical Report. Non-Volatile Memory Technology Center, Austin, Texas.

W. Li, Z. Wasson, and S. A. Seshia. 2012. Reverse engineering circuits using behavioral pattern mining. In *Hardware-Oriented Security and Trust (HOST), 2012 IEEE International Symposium on*. IEEE, 83–88.

W. Li et al. 2013. WordRev: Finding word-level structures in a sea of bit-level gates. In *Hardware-Oriented Security and Trust (HOST), 2013 IEEE International Symposium on*. IEEE, 67–74.

Jim Lipman. 2014. Why Replacing ROM with 1T-OTP Makes Sense. (08 2014). http://www.chipestimate.com/tech-talks/2008/03/11/Sidense-Why-Replacing-ROM-with-1T-OTP-Makes-Sense.

Virage Logic. 2014. Reverse Engineering Techniques in CMOS Based Non-Volatile Memory (NVM). (08 2014). http://www.flashmemorysummit.com/English/Collaterals/Proceedings/2009/20090811_F1A_Zajac.pdf.

Harold G. Longbotham, Ping Yan, Hemal N. Kothari, and Jun Zhou. 1995. Nondestructive reverse engineering of trace maps in multilayered PCBs. In *AUTOTESTCON'95. Systems Readiness: Test Technology for the 21st Century. Conference Record*. IEEE, 390–397.

Roel Maes, Dries Schellekens, Pim Tuyls, and Ingrid Verbauwhede. 2009. Analysis and design of active IC metering schemes. In *Hardware-Oriented Security and Trust, 2009. HOST'09. IEEE International Workshop on*. IEEE, 74–81.

Ruzinoor C. Mat, Shahrul Azmi, Ruslizam Daud, Abdul N. Zulkifli, and Farzana K. Ahmad. 2006. Morphological Operation on Printed Circuit Board (PCB) Reverse Engineering using MATLAB. (2006).

Ian McLoughlin. 2008. Secure embedded systems: the threat of reverse engineering. In *Parallel and Distributed Systems, 2008. ICPADS'08. 14th IEEE International Conference on*. IEEE, 729–736.

Hannes Mio and Franz Kreupl. 2008. IC chip with nanowires. (March 4 2008). US Patent 7,339,186.

Amir Moradi, Markus Kasper, and Christof Paar. 2012. Black-box side-channel attacks highlight the importance of countermeasures. In *Topics in Cryptology–CT-RSA 2012*. Springer, 1–18.

Amir Moradi et al. 2011. 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*. ACM, 111–124.

B. N. Naveen and K. S. Raghunathan. 1993. An automatic netlist-to-schematic generator. *IEEE Design & Test of Computers* 10, 1 (1993), 36–41.

Nikon. 2013. Microscopy. (2013). http://www.microscopyu.com/.

Jean-Baptiste Note and Éric Rannaud. 2008. From the bitstream to the netlist.. In *FPGA*, Vol. 8. 264–264.

NREL. 2014. Scanning Kelvin Probe Microscopy. (08 2014). http://www.nrel.gov/pv/measurements/scanning_kelvin.html.

Xiaochuan Pan. 1998. Unified reconstruction theory for diffraction tomography, with consideration of noise control. *JOSA A* 15, 9 (1998), 2312–2326.

Matteo Patelmo and Bruno Vajana. 2001. Mask programmed ROM inviolable by reverse engineering inspections and method of fabrication. (2001).

Gregory Phipps. 2005. Wire Bond vs Flip Chip Packaging. *Advanced Packaging Magazine* (2005).

Space Photonics. 2013. About Anti-Tamper Technology. (2013). http://www.spacephotonics.com/Anti_Tamper_Reverse_Engineering_Protection.php.

Purdue.edu. 2014. Scanning Electron Microscope. (2014). http://www.purdue.edu/ehps/rem/rs/sem.htm.

Gregory A. Quirk. 2013. Under The Hood: TI 65-nm chip at heart of Nokia 2610. (2013). http://www.eetimes.com/document.asp?doc_id=1281289.

William Radovich and Spencer Worms. 2014. Reverse engineering of integrated circuits. (04 2014).

M. Rahman et al. 2014. CSST: Preventing distribution of unlicensed and rejected ICs by untrusted foundry and assembly. In *Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT), 2014 IEEE International Symposium on*. IEEE, 46–51.

Jeyavijayan Rajendran, Youngok Pino, Ozgur Sinanoglu, and Ramesh Karri. 2012. Security analysis of logic obfuscation. In *Proceedings of the 49th Annual Design Automation Conference*. ACM, 83–89.

Jeyavijayan Rajendran, Michael Sam, Ozgur Sinanoglu, and Ramesh Karri. 2013. Security analysis of integrated circuit camouflaging. In *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*. ACM, 709–720.

Jarrod A. Roy, Farinaz Koushanfar, and Igor L. Markov. 2008a. EPIC: Ending piracy of integrated circuits. In *Proceedings of the conference on Design, automation and test in Europe*. ACM, 1069–1074.

Jarrod A. Roy, Farinaz Koushanfar, and Igor L. Markov. 2008b. Protecting bus-based hardware IP by secret sharing. In *Proceedings of the 45th annual Design Automation Conference*. ACM, 846–851.

Sharedresources. 2014. Transmission Electron Microscope. (2014). http://sharedresources.fhcrc.org/services/transmission-electron-microscopy-tem.

Siliconfareast. 2013. Wet Etching Recipes. (06 2013). http://www.siliconfareast.com/etch_recipes.htm.

Sergei Skorobogatov. 2005a. Data remanence in flash memory devices. In *Cryptographic Hardware and Embedded Systems–CHES 2005*. Springer, 339–353.

Sergei Skorobogatov. 2005b. Semi-invasive attacks - a new approach to hardware security analysis. (2005).

Bernd Stamme. 2014. Anti-fuse memory provides robust, secure NVM option. (08 2014). http://www.eetimes.com/document.asp?doc_id=1279746.

Francois-Xavier Standaert. 2008. Secure and Efficient Implementation of Symmetric Encryption Schemes using FPGAs. *Cryptographic Engineering* (2008), 295.

François-Xavier Standaert, Tal G Malkin, and Moti Yung. 2006. A formal practice-oriented model for the analysis of side-channel attacks. *IACR e-print archive* 134 (2006), 2006.

Stanford.edu. 2014. Transmission Electron Microscope. (2014). http://web.stanford.edu/group/snl/tem.htm.

Pramod Subramanyan et al. 2013. Reverse engineering digital circuits using functional analysis. In *Proceedings of the Conference on Design, Automation and Test in Europe*. EDA Consortium, 1277–1280.

Swarthmore.edu. 2005. CMOS inverter cross section. (2005). http://www.sccs.swarthmore.edu/users/06/adem/engin/e77vlsi/lab3/.

Pawel Swierczynski, Amir Moradi, David Oswald, and Christof Paar. 2013. Physical Security Evaluation of the Bitstream Encryption Mechanism of Altera Stratix II and Stratix III FPGAs. *ACM Trans. Reconfig. Technol. Syst.* (2013).

SypherMedia. 2012. Circuit Camouflage Technology. (2012). http://www.smi.tv/solutions.htm#clo.

Christopher Tarnovsky. 2010. Deconstructing a 'Secure' processor. (02 2010). https://www.youtube.com/watch?v=w7PT0nrK2BE.

Techinsights. 2014. Sony Xperia Play Teardown and Analysis. (2014). http://www.techinsights.com/teardowns/sony-xperia-play-teardown/.

Priya Thanigai. 2014. Introducing advanced security to low-power applications with FRAM-based MCUs. (08 2014). http://www.ecnmag.com/articles/2014/03/introducing-advanced-security-low-power-applications-fram-mcus.

Kris Tiri and Ingrid Verbauwhede. 2004. A Dynamic and Differential CMOS Logic Style to Resist Power and Timing Attacks on Security IC's. *IACR Cryptology ePrint Archive* 2004 (2004), 66.

Randy Torrance and Dick James. 2009. The state-of-the-art in IC reverse engineering. In *Cryptographic Hardware and Embedded Systems-CHES 2009*. Springer, 363–381.

J. A. Van Geloven, P. T. Tuyls, R. A. Wolters, and N. Verhaegh. 2012. Tamper-resistant semiconductor device and methods of manufacturing thereof. (March 27 2012). US Patent 8,143,705.

Steve H. Weingart. 2000. Physical security devices for computer subsystems: A survey of attacks and defenses. In *Cryptographic Hardware and Embedded Systems - CHES 2000*. Springer, 302–317.

Wikipedia.org. 2010. CMOS inverter cross section. (12 2010). http://en.wikipedia.org/wiki/CMOS.

Wikipedia.org. 2014. Printed-Circuit-Board. (2014). http://en.wikipedia.org/wiki/Printed_circuit_board.

Thomas Wollinger, Jorge Guajardo, and Christof Paar. 2004. Security on FPGAs: State-of-the-art implementations and attacks. *ACM Transactions on Embedded Computing Systems (TECS)* 3, 3 (2004), 534–574.

Wen-Yen Wu, Mao-Jiun J. Wang, and Chih-Ming Liu. 1996. Automated inspection of printed circuit boards through machine vision. *Computers in industry* 28, 2 (1996), 103–111.

J. Wu et al. 2010. 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*. ACM, 459–464.

Bulent Yener. 2014. CSCI 4974 / 6974 Hardware Reverse Engineering. (2014). http://security.cs.rpi.edu/courses/hwre-spring2014/.

Frank Zachariasse. 2012. Semiconductor device with backside tamper protection. (June 12 2012). US Patent 8,198,641.

Daniel Ziener, Stefan Aßmus, and Jürgen Teich. 2006. Identifying FPGA IP-Cores based on lookup table content analysis. In *Field Programmable Logic and Applications, 2006. FPL'06. International Conference on*. IEEE, 1–6.