CHAPTER

11 ATTACKS ON PCB: SECURITY CHALLENGES AND VULNERABILITIES

CONTENTS

11.1 INTRODUCTION

Modern PCBs typically integrate a number of ICs with high pin complexity and large number of passive components into a miniature layout [1]. Survey results show that 14% of today's PCBs are currently operating in the 1–10 GHz frequency range to support high-speed data communication [2].

Hardware Security. https://doi.org/10.1016/B978-0-12-812477-2.00016-2

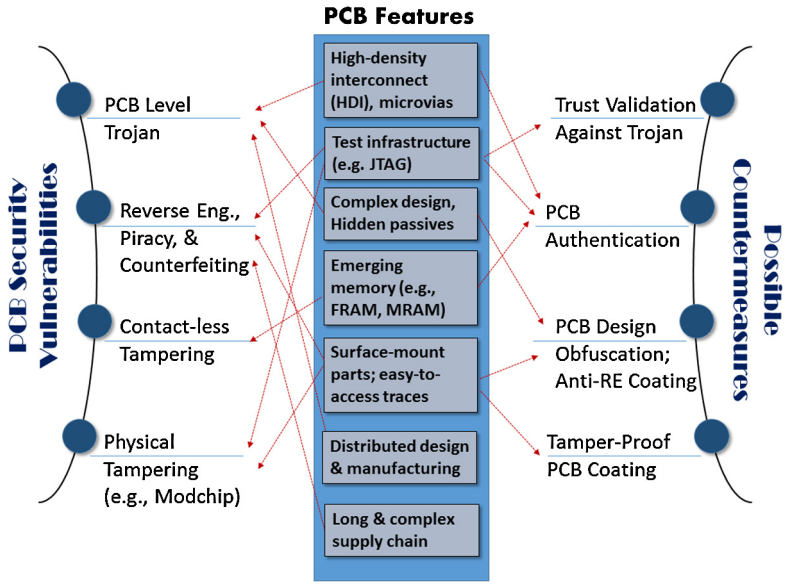

FIGURE 11.1

An illustration showing how the features of modern PCBs create new vulnerabilities. It also shows possible countermeasures, some of which benefit from these features.

The complexity and cost of PCB design are also rising rapidly. With increasing complexity of PCBs—including high-density interconnects, hidden vias, passive components in internal layers, and multiple layers (6–20 layers)—system integrators are increasingly relying on third-party PCB manufacturers. Moreover, the long and distributed supply chain of PCBs is becoming highly vulnerable to diverse attacks that compromise PCB integrity and trustworthiness. A PCB can be deliberately tampered by an adversary through insertion of malicious components, or targeted design changes, that can trigger a malfunction or leak secret information after deployment. Otherwise, these compromised PCBs may suffer from significant performance and reliability issues [3]. On the other hand, counterfeiting has become a major concern in the PCB industry. Counterfeit PCBs pose a major threat in mission-critical systems with serious potential consequences during field operation. Figure 11.1 shows some salient features of modern PCB, arising from the current trend in PCB design, manufacturing, and distribution, and the corresponding security vulnerabilities. It also shows a set of countermeasures at different stages of PCB life cycle (design and test) that one can employ to address these threats. Some of these PCB features can be leveraged to built the countermeasures, e.g., the JTAG infrastructure can be used in trust validation and PCB authentication, as shown in the figure.

Hardware Trojan attacks at the IC level have been extensively studied in recent times. Researchers have analyzed the impact of these attacks and explored possible countermeasures. However, vulnerability with respect to hardware Trojan attacks at higher levels, in particular at PCB level, have not been widely explored. Previous studies have covered security of PCBs against piracy and various post-fabrication tampering attacks. JTAG and other field programmability features in a PCB, for example,

probe pins, unused sockets, and USB have been extensively exploited by hackers to gain access to internal features of the designs, snoop secret key, collect test responses, and manipulate JTAG test pins. One instance of such attack demonstrated that a Xbox gaming console can be hacked by using JTAG to disable the DRM protection. Modern PCBs are becoming increasingly vulnerable to malicious modification of PCBs during design or fabrication in untrusted design or fabrication facilities. Such a vulnerability creates a new class of threat for PCBs. The emerging business model of PCB design and fabrication, which favors extensive outsourcing and integration of untrusted components/entities in the PCB lifecycle to lower manufacturing cost [4–6], makes hardware Trojan attacks in PCBs highly feasible.

A closer look at several major electronic products and their PCB manufacturers reveals that PCBs are often designed in various countries. Moreover, reliance on third-party manufacturing facilities makes the PCB fabrication process untrustworthy and, hence, vulnerable to malicious modifications. Furthermore, an adversary can be present inside the design house and can implant a Trojan into a PCB design. PCBs in today's complex and highly integrated designs contain as many as 20 to 30 layers with hidden vias and embedded passive components [7] to minimize the form factor. This presents a great opportunity for an attacker to deliberately modify a PCB design by tampering the interconnect lines at the internal layers, or altering the components.

Due to the highly distributed nature of PCB design flow, maintaining high levels of security standard across the entire supply chain system has become a challenging task. Consequently, the weak links of the PCB life cycle are more vulnerable to security breaches and malicious attacks by rogue entities. Incorporation of untrusted vendors across the globe is exacerbating the situation by introducing novel threats in PCB life cycle and supply chain system. Despite the associated risks, the manufacturing companies are being compelled to adopt the existing horizontal business model for PCB production to cope up with changing landscapes of the semiconductor industry and reduce design, and manufacturing cost.

The relative ease of reverse engineering a PCB (compared to IC) using home-based solutions has also been reported in the literature. An adversary can steal a PCB design or reverse engineer a fabricated PCB to obtain the design information. Next, he/she can make pirated and counterfeit copies, or resell reverse-engineered PCB designs. Moreover, it is possible to extract the vulnerable points and launch cleverly crafted attacks on PCBs. A broad classification of the attacks on PCBs is illustrated in Fig. 11.2. The primary categories in the taxonomy include piracy and counterfeiting issues, hardware Trojan attacks, and in-field alteration. In this chapter, each attack category is discussed in detail. We present relevant case studies to further illustrate the attacks and vulnerabilities. The limitations of traditional PCB testing in verifying the security and trustworthiness of a PCB are also discussed. Finally, an in-field alteration attack, referred to as Modchip attack, is explained with an example attack launched on commercial Xbox gaming console [10–12].

11.2 PCB SECURITY CHALLENGES: ATTACKS ON PCB

In this section, we describe some of the attacks on PCB in detail.

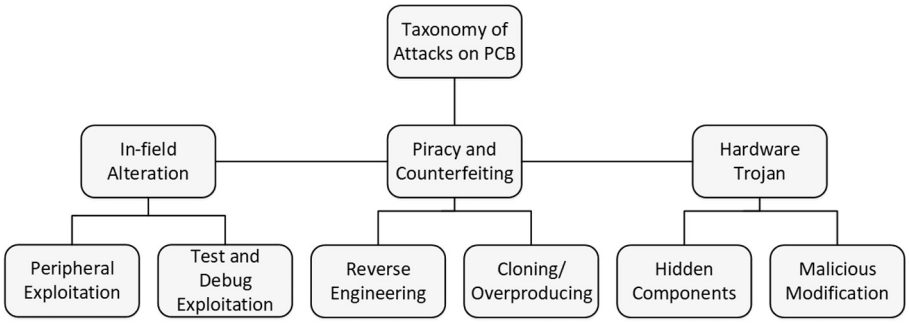

FIGURE 11.2

A taxonomy of attacks on PCBs.

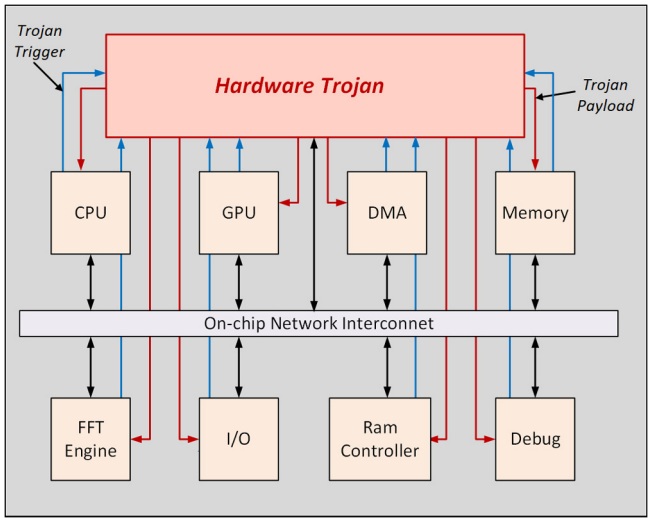

FIGURE 11.3

A generic overview of hardware Trojan at PCB level.

11.2.1 HARDWARE TROJANS IN PCB

A generic model of hardware Trojan attack on a PCB is illustrated in Fig. 11.3. At PCB level, hardware Trojans may have two types of payload. First, Trojans can interrupt or maliciously change the functionality of a PCB, making it fail during field operation. For example, addition of a capacitor on PCB signal lines can cause disruption in the regular circuit operation and communication among the components in the board. Such alteration can lead to in-field failure. Second, a Trojan can leak sensitive information from a PCB design. An example of such attack would be inserting a capacitor-based leakage circuitry to extract critical system information, such as keys of cryptographic modules.

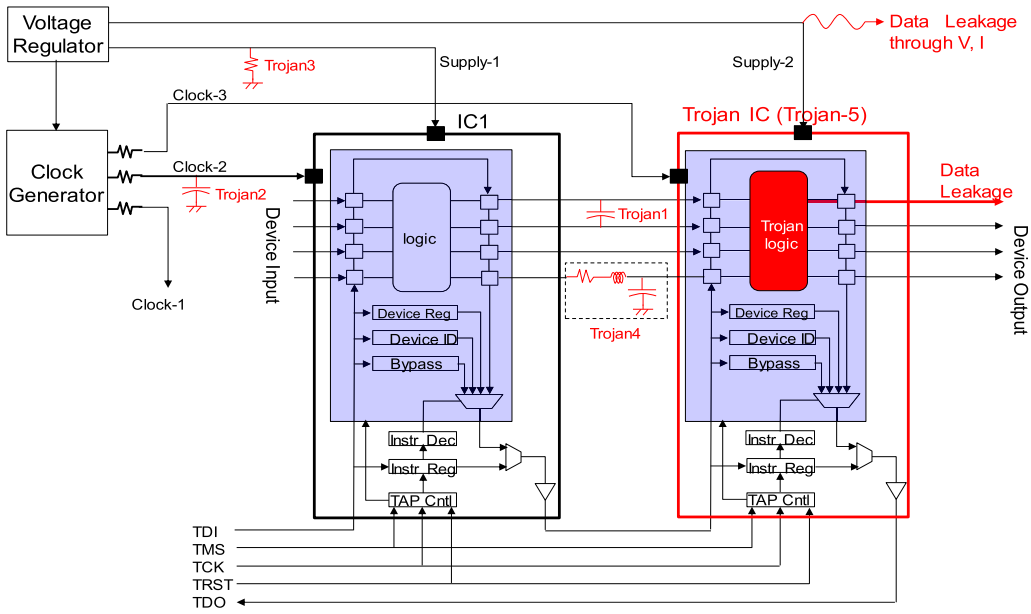

FIGURE 11.4

An illustration of hardware Trojan insertion into a PCB as a hidden component.

11.2.1.1 Hidden Components

In a multilayer PCB board, an adversary can insert additional electrical components in one of the layers to obtain secret information by leaking it through various means. Similarly, an attacker can cause malfunctioning or operate the PCB beyond its regular specifications by inserting malicious components in the original design. One instance of such attack would be, replacing the original IC with a tampered, counterfeit, or custom-designed IC containing hardware Trojans. The new IC may have nearly equivalent functional and performance specifications, making the Trojan hard to detect using conventional PCB testing. An illustrative example of such Trojan in PCB-level design is provided in Fig. 11.4.

11.2.1.2 Malicious Modifications

An adversary may also modify the resistance, inductance, or capacitance values of signal traces of a PCB. An example of such modification is reduction of the width of an internal layer trace to increase its resistance. Consequently, this alteration may lead to design failure over a longer period of operation due to overheating. Similar attacks could be employed to change the coupling capacitance of the traces leading to delay failure. Introducing additional coupling voltage in the circuitry requires modifications, such as altering inter-trace distance through re-routing, and selectively changing dimensions of traces and dielectric properties. Attacks can be mounted to degrade operating voltage in a PCB by incorporating high-resistance paths in the internal layers. The interconnection between two components can also be corrupted to disrupt the design functionality by introducing impedance mismatch. To evade

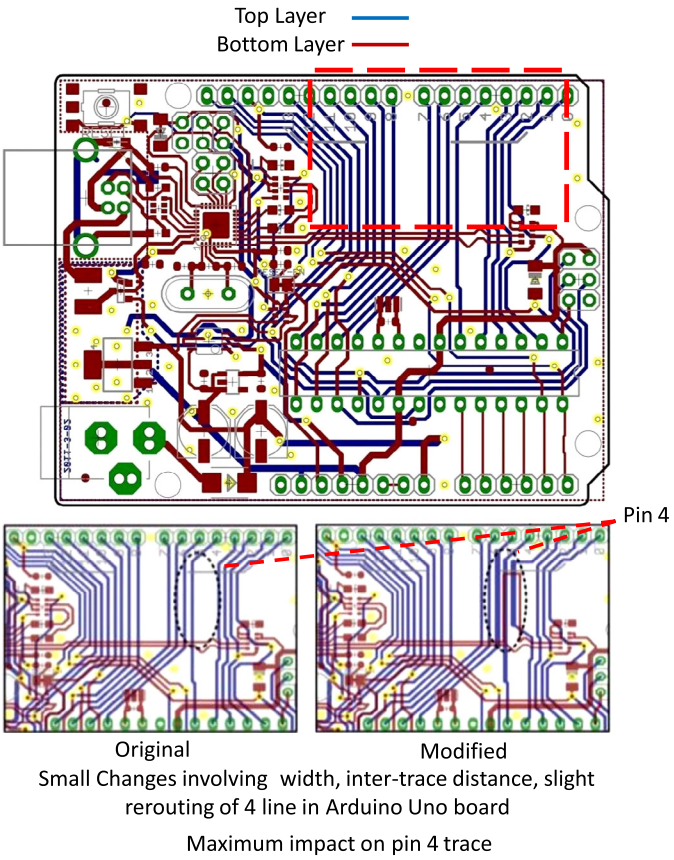

FIGURE 11.5

Minor modifications in an Arduino UNO PCB layout to insert a hardware Trojan without addition of any new component.

conventional PCB testing, these Trojans need to be triggered by rare internal conditions in a PCB or through external trigger mechanisms, which are difficult to exercise during test.

An illustrative example of malicious trace modification attack on PCB is demonstrated on a commercial Arduino Uno board (Fig. 11.5). The attack shows a possible approach for tampering trace lines in PCBs. The major impacts of such tampering are: degradation of the output voltage and circuit failure caused by delay, or additional coupling voltage. It involves altering the trace thickness and inter-trace distance by 2X of the original design, and rerouting a single trace. Note that the modification is difficult to trace via traditional testing approaches, though it is incorporated on a relatively simple two-layer PCB. An increase in the design complexity with additional layers would enhance the chances of the alteration to evade visual-inspection-based testing, such as optical or X-ray imaging. Moreover, it is generally hard to detect these modifications via functional and parametric testing of PCB, as these tests are applied for verifying the limited functionality of the board. It is not a feasible option to adopt

exhaustive testing methodology due to time and resource constraints. Figure 11.5 depicts the modified trace lines and the impact of the modification. Note that the voltage degradation at Pin 4 is expected to lead to board malfunction during field operation.

11.2.2 IN-FIELD ALTERATION

11.2.2.1 Peripheral Exploitation

Peripheral exploitation can be defined as an attempt made by an attacker to exploit on-board ICs and other electrical components (active and passive) to launch an attack. Common instances of peripheral exploitation include mounting rogue ICs on the original design, changing the connection of wires via soldering, rerouting the circuit data path to evade or substitute a security block, or access restricted block on the PCB. An example of such attack is Modchip attack [11]. Modchips represent a unique class of components capable of maliciously changing the function of a PCB. They are also used to limit access to restricted part of a design. Modchips are often mounted on set-top boxes and gaming consoles for manipulating the built-in protection in these devices [13].

11.2.2.2 Test and Debug Exploitation

PCB design features, such as JTAGs, USBs, test pins, and test/debug structures can be exploited as attack surfaces by an adversary. An attacker can exploit these features to understand the intent of a design, and mount attacks more efficiently with minimal alterations.

- **JTAG interface:** As discusses in Chapter 4, JTAG is an industry standard developed to test and debug PCBs after fabrication. JTAG integrates several test features of the board for the ease of testing and debugging. For example, JTAG has access to data and address bus of onboard chips for testing functionality and performance. Such accessibility, however, can be exploited by an attacker to retrieve sensitive design information or stored secrets in a PCB. An attacker can attempt to gain control over chip data and address buses of the board to mount an attack on the data bus by exploiting JTAG. The attack requires acquiring information about the instruction registers, such as the size and functionality of the registers through a trial-and-error method. Once the relevant information is obtained, specific instructions can be executed to get access to the system data, and feed the bus with corrupted data. Another example of JTAG attack would be reverse engineering the design via connectivity inspections of components onboard.
- **Test pins or probe pads.** Most ICs are designed with probe pads and test pins to observe and control important signals for test and debug purposes. An adversary can tap these pins and monitor the critical signals to gain information about the functionality of the design, or feed malicious data into the design. Test pins can also be exploited for reverse engineering, where a test input can trigger certain data, address and control signals that can help identify the board functionality. A list of additional vulnerabilities originating from common PCB design features is provided in Table 11.1.

11.2.3 PIRACY AND COUNTERFEITING

11.2.3.1 Reverse Engineering

PCBs are highly vulnerable to reverse engineering. An adversary can purchase a PCB, or a system containing a PCB, from the market, and attempt to reverse engineer it. It has been demonstrated in prior work that even a complex multilayer PCB can be completely reverse engineered in a relatively

Table 11.1 Various attack surfaces created by PCB design features

Security vulnerability in PCB	Possible exploitation for hardware / physical attacks
JTAG	Control of scan chain/data bus for illegal access to memory/logic
RS232, USB, Firmware, Ethernet	Access to internal memory of an IC, leaking secret data
Test Pins	Access to internal scan chain to leak data from ICs
Unused pins, multiple layers, hidden vias	Alteration of connections using internal layers/hidden vias

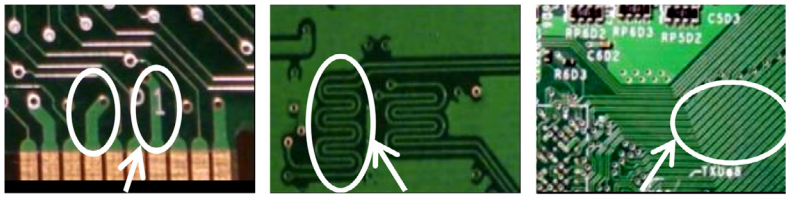

FIGURE 11.6

Visual inspection of PCB can reveal critical design information: (A) Thicker traces for high-frequency signals; (B) Pair of signals for differential signaling; (C) Group of traces indicating bus.

simple manner with low-cost home-based solutions. A reverse engineered PCB can then be cloned to produce unauthorized copies. PCB reverse engineering may also help an adversary to better understand the design and then tamper it effectively. An adversary can create counterfeit copies of a PCB after reverse engineering the design. It is generally easy to assemble counterfeit PCBs since most PCBs use active/passive components that are readily available in the market. These PCBs can simply be low-quality fakes or can include malicious circuits, i.e., hardware Trojan, as discussed before [8]. Finally, the process of reverse engineering, may enable an attacker to extract critical security information of a design; identify any vulnerability therein; and then develop powerful attacks on the system [9].

11.2.3.2 Cloning

Attackers can clone original PCBs with malicious intents. Visual inspection of the PCB boards can reveal critical information about the design and facilitate the PCB cloning process. Illustrative examples of such scenarios are depicted in Fig. 11.6. The description of each type of vulnerability is as follows:

- **Distinct properties of special signals:** An adversary can guess the functionalities of different signals by their distinct properties. For instance, the thickness of the trace and the group of traces of a data bus provide clues about the functionality. Similarly, pins tied with identical pull-up/down resistors indicate that they belong to a bus.
- **Remnant signatures from test or debug:** When the test and debug pins are accessed through ports, the remnant of soldering provides intuitive clues about the functionality of these pins. An empty socket on the PCB can also be exploited by an adversary for mounting an attack.
- **Miscellaneous hints:** Apart from the attack surfaces provided by component-level hooks, a PCB design itself reveals lots of information to an adversary in fabrication house that can facilitate pow-

erful Trojan attacks. Figure 11.6 depicts how traditional design features and miscellaneous hints can be exploited by attackers to comprehend design functionality.

11.3 ATTACK MODELS

Trojan attacks in PCB can be divided into two broad classes as described below.

Case 1: PCB design is trusted. In this attack scenario, it is assumed that the designs are obtained from trusted parties. The fabrication facility is deemed untrustworthy and flagged as possible source of an attack. Moreover, it is assumed that an intelligent attacker is capable of evading conventional post-manufacturing tests. The goal of the attacker is to trigger the attack in a rare situation (that is, involving a rare combination of inputs) that is difficult to check via regular functional or parametric testing.

Case 2: PCB design is not trusted. The threat model for this instance considers both the board design and fabrication facility to be untrusted. Only the functional and parametric specifications of the board are trusted. In this instance, an attacker has a higher flexibility of maliciously altering the design and/or choosing fake or untrustworthy (and potentially malicious) components. Again, an attacker would try to hide the modifications to avoid detection during functional and parametric testing process.

Note that in both cases, there are two possible objectives of the attacker: 1) malfunction, and/or 2) information leakage. The possible Trojan attacks of different forms on a PCB are described in the following sections.

11.3.1 ATTACK INSTANCES

11.3.1.1 Design House is Trusted

This attack arises when the PCB is designed by a trusted designer and outsourced for fabrication. During the manufacturing process, it is possible for the adversary to insert malicious modifications intelligently, such that the final design structurally matches the original one. In such case, no additional components—such as logic and traces—are integrated but the design will produce undesired functionality under certain conditions. The goals for altering the existing traces can be: increasing the mutual coupling capacitance, characteristic impedance, or loop inductance by changing internal layer routing and inserting small leakage paths. Additional components with an ultra-low area and power requirements can also be inserted in the internal layers. The small alterations can be confined to the internal layers of a multilayer PCB. Hence, chances of detection with visual inspection, optical imaging, and x-ray based imaging techniques are low. Also, it is infeasible to perform exhaustive functional testing with a large number of test nodes. Consequently, the malicious functions are very unlikely to get triggered during the in-circuit and boundary-scan-based functional testing. Additional components with ultra-low area and power requirements can also be inserted in the internal layers.

In this section, two Trojan examples are presented as attack instances. In the first case, a multilayer PCB (10 cm length) is considered with possible application in a high-speed communication and video streaming systems. In this board, there are two high-frequency (HF) PCB traces in an internal layer running parallel to each other. Usually, HF traces are routed in the internal layer, shielded by power and ground planes to avoid interference (Fig. 11.7A). However, the procedure significantly complicates

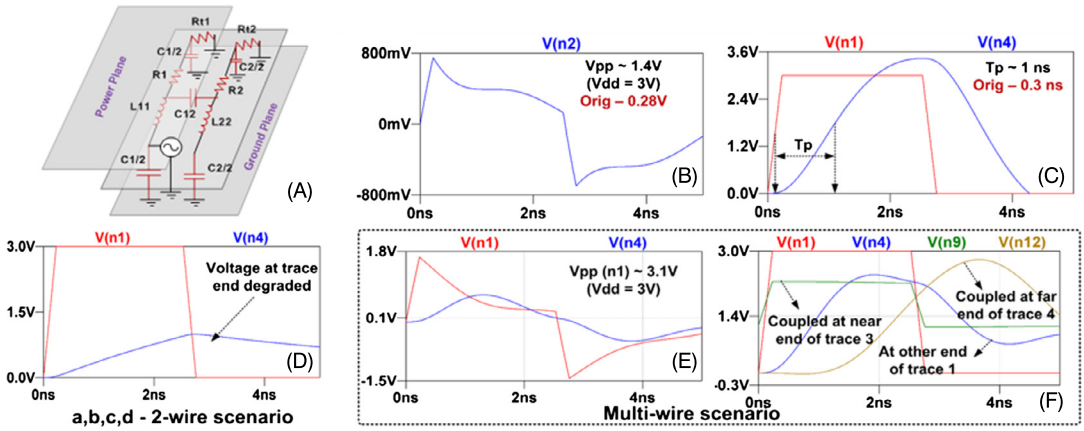

FIGURE 11.7

This set of figures illustrates the results of PCB level Trojan attacks. In this attack scenario, specific traces and properties of the PCBs are altered without introducing any new component into the original design. The figures are depicted as follows: (A) A lumped two-trace PCB system with associated components, that is, resistors, capacitor, and inductors; (B) The near and far-end voltages of trace-2; (C) The propagation delay in trace 1 (at node n1 and n4) at 220 MHz with an input voltage of 3 V peak-to-peak; (D) Insertion of a leakage resistance path from trace 1 to ground changes the far-end voltage of the respective trace; (E) and (F) illustrate how the trace property changes in a 4-wire scenario. In particular, it depicts the effect of coupled voltages in the near and far-end of victim traces, while all aggressors are switching in phase with a frequency of 220 MHz and a peak-to-peak voltage of 3 V; (F) shows the voltage profile at node 9, that is, the near-end of trace 3 and node 12, that is, the far-end of trace 4.

the internal layer testing and debugging, and creates an attack surface for the attacker. The dimensions of the traces are chosen carefully to carry normal HF signals, that is, 1 ounce copper trace with width and thickness of 6 mils and 1.4 mils, respectively. The dielectric is FR-4 with a relative permittivity of 4.5. The inter-trace distance is chosen to be 30 to 40 mils to avoid the negative effects of mutual inductive and capacitive coupling. These HF traces are modeled by lumped parametric form. Functional simulation results show a maximum coupled near- and far-end voltage of ∼300 mVpp on one of the traces. The other trace is swept with pulse voltages of 3 Vpp at 10–500 MHz, with a 50% duty cycle. The maximum propagation delay for the pulse across the active trace is ∼0.4 ns.

Following are the effects of various trace level modifications we observed during fabrication with the aforementioned setup: the inter-trace distance in the internal layers reduced by 2x; widths of both wires increased by 2x, and the thickness increased by 1.5x. Due to the minimal nature of the changes in a small target region of an internal layer, these manipulations are mostly undetectable during structural testing. The dielectric permittivity of the insulator between the traces increased to 5.5 to model moisture retention in certain insulating areas, add impurity to epoxy base and hence, facilitate the aging effect. As the permittivity is selectively altered by an adversary in a small area, the accelerated aging tests have a low probability of detecting the change. However, the impact of these changes on associated circuit parameters can be significant. At 220 MHz, the near-end peak-to-peak voltage in trace(2) is ∼1.4 V for an input pulse voltage of 3 Vpp in trace(1) (Fig. 11.7B). This is an extraneous interference and may

cause unexpected behavior in terms of erroneous circuit activation or feedback. The propagation delay increases by 2x, beyond 1 ns (Fig. 11.7C), which can induce functional failures for higher switching frequencies and greater trace lengths. An attacker can insert and exploit a leakage path to drain the target signals via ground. As a result, there is voltage degradation, which is demonstrated in Fig. 11.7D by plotting the distorted waveforms at the far end of trace(1). The ultimate goal of the attacker is to cause circuit malfunction through severe voltage degradation. This attack can easily evade detection by conventional PCB testing, as these are not exhaustive, due to prohibitive cost and time-to-market requirements.

One strategy to enhance mutual coupling is intentional rerouting of multiple HF traces. If the process is adapted for different planes, the effect of coupling becomes even more prominent. This phenomenon can be observed by minimizing the distance between traces located in the same plane, and increasing the thickness and widths of trace lines. It is highly unlikely that these minute changes will get noticed during the structural and functional tests. The consequence of these alterations, however, can be quite significant on circuit performance, as demonstrated in Fig. 11.7E–F. The resultant coupling voltages measured at the near- and far-end of the target trace were 3.1 V and 1.3 V, respectively. It should be noted that these were peak-to-peak voltage values with in-phase rising/falling transition on the three adjacent traces (one in-plane, one above, and one below) (Fig. 11.7E). This is 3 to 4 times greater than the scenario when active traces were switching in opposite directions. Such interference could certainly lead to failure situations, such as erroneous activation, feedback, and degraded circuit performance. The voltage profile at the far end of trace demonstrated some distortions along with an average propagation delay of 1 ns, while other traces were inactive (Fig. 11.7F). The propagation delay increased with the increasing number of neighboring traces and lengths of the traces. This could lead to delay failures at operations with high switching speed. Extraneous coupled voltages for traces three and four are delineated in Fig. 11.7F. The primary observation from the results is that these Trojan attacks via alteration of traces are extremely hard to detect, as they are sensitized in a very rare set of conditions. In case of multi-wire scenario, the degraded performance was significant only in two out of eight possible combinations of transition polarity (that is, all rising/falling pulses) in three neighboring PCB traces. The frequency of operation of the system and the input vector patterns delivered by selective trace properties and routing alterations made during PCB fabrication were exploited as the triggering condition of the Trojan.

11.3.1.2 Design House is Untrusted

The combination of the untrusted design house and foundry manifoldly increases the vulnerability of Trojan attack. For this instance, the system designer is assumed to be a trusted entity. The primary task of the system designer is to verify the functionality and performance of the design through post-manufacturing PCB testing. In this attack scenario, the capability of the attacker is not limited to trace-level modifications. Access to untrusted foundry opens the opportunity for an attacker to modify the design structurally, and integrating additional malicious components that can get triggered by a predetermined set of conditions. Intelligently designed Trojan can be made stealthy by making it difficult to detect through optical inspection and setting the trigger condition to rare internal signal states. To further illustrate the attack scenario, a Trojan attack is demonstrated through a microcontrolled fan-speed controller. The controller works with a 12-V brushless DC fan that relies on the inputs of a temperature sensor. Based on the temperature reading, the sensors deliver an output voltage varying over 0 to 5 V. The voltage values are digitized by an ADC (analog-to-digital converter) and sent to the

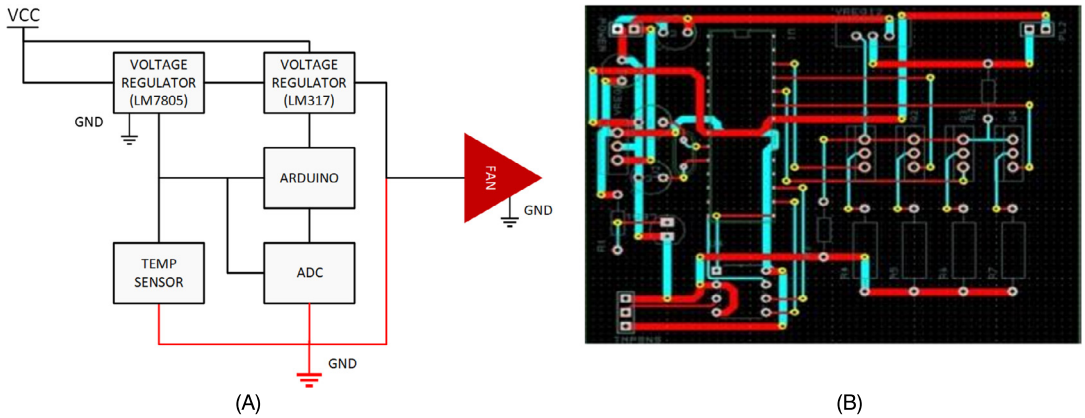

FIGURE 11.8

Example attack scenario when design house is untrusted: (A) Trojan inserted fan controller circuit; (B) A 2-layer PCB layout of the original circuit.

microcontroller to adjust the fan speed accordingly. The adjustment is done via linear regulation of fan input voltage.

Hardware Trojan attacks aiming at deliberate changes in the system functionality can be mounted via subtle structural modification of the PCBs. In this instance, it is possible to hamper the correct functionality of a microcontroller by launching a Trojan attack. Figures 11.8A and 11.9 refer to such an attack instance. In this attack scenario, the microcontroller contains three electrical components, that is, a PMOS transistor, a resistor, and a capacitor. The system is designed in a manner that the capacitor gets charged through the output of a specific voltage regulator, that is, LM317 connected to the fan circuitry. The attacker aims at triggering the Trojan by specific sets of values obtained from the resistor and the capacitor. The timing of the Trojan activation can also be fine-tuned via manipulation of the capacitive and resistive values. Once the Trojan is triggered, it nullifies the functionality of the PMOS transistor located in between the ADC and the temperature sensor. Consequently, the microcontroller will deem the null input as a temperature value of very low range, and reduce the fan speed significantly. Such failure, inaccurate temperature detection in mission-critical devices, may lead to catastrophic consequences. Moreover, the aforementioned Trojan can easily evade the functional testing phases by applying a large value of a time constant. The target design, a 2-layer PCB designed for microcontroller fan system, is illustrated in Fig. 11.8B. The figure depicts the design in pre-fabrication stage. Figure 11.9 demonstrates the Trojan in the fabricated PCB, the triggering of the attack, and the payload deliverance.

11.3.2 IN-FIELD ALTERATION

It is possible for an adversary to launch an attack on a PCB irrespective of its origin; a trusted or untrusted design house, through in-field physical alterations, if the system allows unauthorized access.

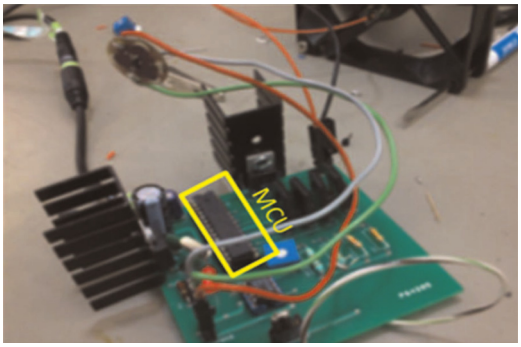

FIGURE 11.9

A fabricated PCB board, demonstrating triggering and payload of a Trojan.

11.3.2.1 Modchip Attack

A dominant, yet, least discussed security threat to the PCB is in-field alteration. The alteration can be caused by mounting ICs, soldering wires, rerouting paths to avoid or substitute existing blocks, adding or replacing components, exploiting traces, ports or test interfaces, and in many other ingenious ways. Circumventing DRM protection by tampering the PCB of a gaming console is a prominent example of PCB tampering. Physical alteration to disable built-in restrictions allows the user to play pirated, burnt, or unauthorized versions of a game on the hacked console. Modchips are devices that are used to alter the functionality or disable restrictions within a system, such as a computer or a video game system. Modchips usually contain a microcontroller, FPGA, or CPLD (complex programmable logic device) in order to attack the host system. They are soldered into the host system on top of the security-critical traces, as can be seen in Fig. 11.10. An industry-standard interface designed by Intel functions through the low-pin-count data bus, as shown in Fig. 11.10. The LPC bus is used for testing and debugging the Xbox during the production phase. Once these devices are installed they are often used for illegal purposes, such as playing illegally copied games and other forms of digital rights violations. For instance, Xbox Modchips can modify or disable the built-in restrictions integrated into Xbox consoles, allowing the users to play pirated games on the tampered consoles. Piracy leads to loss of revenue for game developers, and a reduced budget for future games.

An adversary can perform such tampering on a PCB to bypass DRM key-based protections in various systems. An illustration of such PCB-tampering attack is given in Fig. 11.11. The figure shows an attack instance, where the channel access grant signal of a TV set-top box is reliant on the DRM keys stored in a nonvolatile memory. A standard comparator is employed to grant the access signal in accordance to the channel number and corresponding DRM key. The comparator delivers a high signal in case of key match, and vice versa. However, it is possible to get access to the channels irrespective of the DRM keys by tampering the access grant signal wire. An attacker can tamper the wire by connecting it to the Vdd signal originating from a voltage regulator. Consequently, the access grant wire will always send a high signal to the channel-control circuitry, and the attacker will be given access to protected channels without the right keys. The tampered PCB trace is highlighted in red (dark gray in print version) to depict the attack mechanism. A protection scheme against such attacks would be integration of additional tamper-detection and protection circuitry for the critical

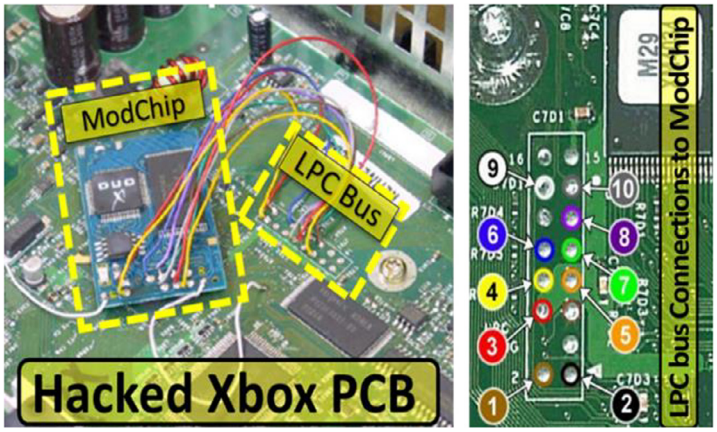

FIGURE 11.10

An illustration of Modchip attack. Physical tampering of the PCB of an Xbox gaming console is depicted in the figures: (A) Modchip wired to PCB of Xbox via low-pin-count (LPC) bus; (B) An illustration of LPC bus with associated pins of Modchip.

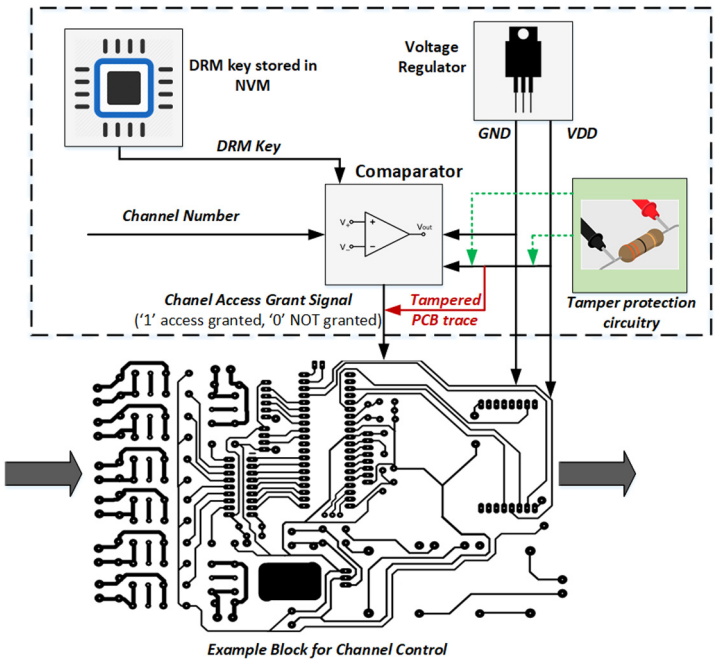

FIGURE 11.11

An illustrative example of Digital Rights Management (DRM) protection bypass through in-field PCB tampering.

traces on a PCB. The green (medium gray in print version) block in the PCB diagram (Fig. 11.11) shows a high-level representation of a resistance-sensing circuitry for the critical traces in the PCB. The physical tamper-detection and protection circuitry consists of a microcontroller and an e-fuse that trips the circuit in case of anomalies detected in resistance of the critical traces. Hardware attacks like Modchip alterations may jeopardize a company's profit margin. Video game piracy through tampered consoles caused around 1.45 billion pound sterling loss in sales in the UK in 2010. This led to around 1000 fewer jobs in the video game industry during that period [11,14]. These attacks also victimize the consumers when the gaming consoles or computing systems are resold in the market after tampering. These counterfeits cannot be trusted for a secure and safe operation. Mobile devices, embedded system, and the IoT devices are also vulnerable to the threats of Modchip attacks. Example of such attacks would be interfering with the data between DRAM, NAND flash, and SoCs via Modchips. Modchips can also be exploited to retrieve and alter data and system codes written from memory to SoC.

11.4 HANDS-ON EXPERIMENT: BUS SNOOPING ATTACK

11.4.1 OBJECTIVE

This experiment is designed to give students exposure to noninvasive bus-snooping attacks in a system. The attack is applied on the HaHa platform. The objective of this experiment is to give students practical experience to snooping through physical PCB-probing techniques.

11.4.2 METHOD

Students have to first map an example design into the microcontroller and the accelerometer on the HaHa platform. Next, the students will place testing probes in different wires between the microcontroller and the accelerometer to capture and observe the data flow. The example design allows the students to control the type of the mapped operation, while analyzing the behavior of each operation.

11.4.3 LEARNING OUTCOME

By performing the specific steps of the experiments, the students will learn how a bus-snooping attack is done, the associated challenges, and tools and techniques that can be used by an attacker to extract critical information from the PCB under attack. They will also experience the opportunities and explore possible solutions with respect to protecting hardware against snooping attacks.

11.4.4 ADVANCED OPTIONS

Additional exploration on this topic can be done through locating and applying the snooping attack to other components and interfaces (for example, processor-memory bus, Bluetooth communication, and interface).

More details about the experiment are available in the supplementary document. Please visit: http://hwsecuritybook.org.

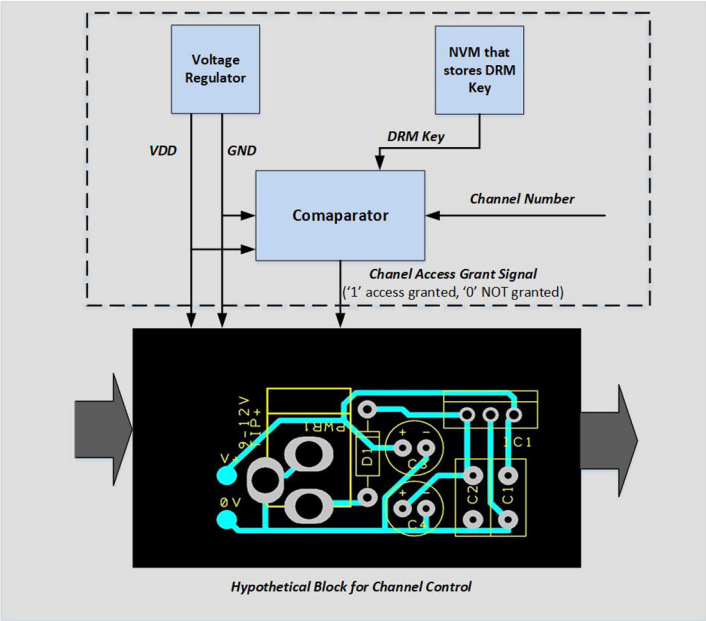

FIGURE 11.12

High-level block diagram of a PCB used in a TV set-top box.

11.5 EXERCISES

11.5.1 TRUE/FALSE QUESTIONS

1. Hardware Trojans in PCB could help cloning the design.
2. Modern-day PCBs are designed with a single layer.
3. JTAGs can be exploited to hack scan-chains.
4. It is not possible to leak information via PCB test pins.
5. Modchip attack is an example of in-field alteration.
6. Only untrusted design houses are vulnerable to PCB attacks.
7. Trusted design houses are not susceptible to malicious attacks.
8. Modchip attacks are prevalent in gaming consoles.
9. PCB reverse engineering requires industry-grade equipment and expertise.
10. It is comparatively easier to find a Trojan in multilayer PCBs.

11.5.2 SHORT-ANSWER TYPE QUESTIONS

1. Define hardware Trojans.
2. What is the definition of a Modchip?

3. What is JTAG?
4. What primary purpose a PCB serves in any electronic hardware system?
5. Describe possible attacks on a PCB.
6. In the high level block diagram of a PCB used in a TV set top box (Fig. 11.12), the Digital Right Management (DRM) key is stored in a non-volatile memory (NVM) which goes to a comparator that generates the channel grant access signal. Describe a possible tampering attack that can bypass this protection. You can update the drawing to illustrate your attack. You need to incorporate your attack within the dotted box.
7. What would be a possible solution to protect against the aforementioned tampering attacks?
8. What is PCB reverse engineering?

11.5.3 LONG-ANSWER TYPE QUESTIONS

1. Describe a potential attack instance when the design house is trusted in the PCB manufacturing process. Illustrate the scenario with an appropriate case study.
2. What would be an attack instance on PCBs manufactured in untrusted design houses? Describe with a relevant case study.
3. What are the possible attacks on PCBs? Explain the taxonomy of attacks with a brief description of each attack-type.
4. Describe a hardware Trojan attack instance on an insecure PCB.
5. How can a Modchip attack be mounted on a PCB? Explain elaborately.

REFERENCES

[1] S. Bhunia, M.S. Hsiao, M. Banga, S. Narasimhan, Hardware Trojan attacks: threat analysis and countermeasures, Proceedings of the IEEE 102 (2014) 1229–1247.
[2] R.S. Chakraborty, S. Narasimhan, S. Bhunia, Hardware Trojan: threats and emerging solutions, in: High Level Design Validation and Test Workshop, 2009. HLDVT 2009. IEEE International, IEEE, pp. 166–171.
[3] Y. Alkabani, F. Koushanfar, Consistency-based characterization for IC Trojan detection, in: Proceedings of the 2009 International Conference on Computer-Aided Design, ACM, pp. 123–127.
[4] H. Salmani, M. Tehranipoor, J. Plusquellic, A layout-aware approach for improving localized switching to detect hardware Trojans in integrated 386 circuits, in: Information Forensics and Security (WIFS), 2010 IEEE International Workshop on, IEEE, pp. 1–6.
[5] S. Ghosh, A. Basak, S. Bhunia, How secure are printed circuit boards against Trojan attacks? IEEE Design & Test 32 (2015) 7–16.
[6] W. Jillek, W. Yung, Embedded components in printed circuit boards: a processing technology review, The International Journal of Advanced Manufacturing Technology 25 (2005) 350–360.
[7] S. Paley, T. Hoque, S. Bhunia, Active protection against PCB physical tampering, in: Quality Electronic Design (ISQED), 2016 17th International Symposium on, IEEE, pp. 356–361.
[8] J. Carlsson, Crosstalk on printed circuit boards, SP Rapport, 1994, 14.
[9] B. Sood, M. Pecht, Controlling moisture in printed circuit boards, in: IPC Apex EXPO Proceedings, 2010.
[10] O. Solsjö, Secure key management in a trusted domain on mobile devices, 2015.
[11] Modchip.net, https://www.mod-chip.net/, 2011. (Accessed 10 September 2018).
[12] D. Whitworth, Gaming industry lose 'billions' to chipped consoles – BBC newsbeat, 2011.
[13] S. Chhabra, B. Rogers, Y. Solihin, SHIELDSTRAP: Making secure processors truly secure, in: IEEE International Conference on Computer Design, 2009.
[14] J. Grand, K.D. Mitnick, R. Russell, Hardware Hacking: Have Fun While Voiding Your Warranty, Syngress, 2004.