硬件木马:关键问题研究进展及新动向

黄 钊 王 泉 杨鹏飞
(西安电子科技大学计算机学院 西安 710071)

摘 要 全球化商业模式下,电子计算机行业的设计人员需要利用不同国家或区域的设计/制造服务以及第三方知识产权(Third-Party Intellectual Property,3PIP)核来完成集成电路(Integrated Circuit,IC)的设计和制造.整个产业链上任何节点的漏洞都有可能成为攻击者实施攻击的入口,而硬件木马作为其中一个突出的安全威胁,已经成为 IC 设计与制造领域一个亟需解决的问题,有必要对其技术特点、研究现状和发展趋势进行梳理总结.该文首先简要介绍了硬件木马的基本概念和相关技术,对硬件木马的国内外研究成果和最新工作进行了整理归纳;讨论了硬件木马研究中的关键问题.该文针对硬件木马的研究热点内容进行了详细的对比总结,总结了当前研究工作的成果及存在的不足,讨论了硬件木马及相关技术的发展趋势.

关键词 硬件安全;硬件木马设计;安全防护技术;评估与分析
中图法分类号 TP303 **DOI 号** 10.11897/SP.J.1016.2019.00993

Hardware Trojan: Research Progress and New Trends on Key Problems

HUANG Zhao WANG Quan YANG Peng-Fei
(*School of Computer Science and Technology, Xidian University, Xi'an* 710071)

Abstract As the design complexity and manufacturing costs of modern electronic devices continue to increase, designers shift toward utilizing the overseas design and manufacturing services and third-party intellectual property (3PIP) cores from different countries or regions to complete the entire design and manufacturing process of the integrated circuit (IC). However, the globalization trend and highly distributed characteristics of current IC design-fabrication flow have incurred the vulnerabilities of modern IC supply chain, which forms new attack sources. In this circumstance, rogue entities could malicious involve within any stage of the overall cycle life of the electronic electronics or ICs, resulting in the designers or customers of ICs begin to raise their doubts about the hardware level security and trustworthy of the products. There are various forms of existing and emerging security attacks in current industrial supply chain. As one of the most prominent security threats in current design and manufacturing area, hardware Trojan attack is now being considered as an urgent problem to be solved in information and hardware security domains. Such threat has aroused widespread concerns in academia and industry. What's more, there have been a lot of survey research work on hardware Trojan and its associated techniques at home and abroad. Unfortunately, these surveys have some limitations more or less. For instance, some work only summarizes the progress of a certain stage or a certain detection approach, and most of the references for those research work were published before 2014, which could not represent the latest research progress and development trend in this area. In particular, with the deepening of

收稿日期:2018-05-21;在线出版日期:2018-09-26.本课题得到国家自然科学基金项目(61572385,61702395,61711530248)、陕西省科技统筹创新工程计划项目(2015KTCXSF-01)资助.**黄 钊**,博士研究生,中国计算机学会(CCF)学生会员,主要研究方向为嵌入式系统开发、硬件安全、硬件木马检测.E-mail:zhhuang@stu.xidian.edu.cn.**王 泉**(通信作者),博士,教授,博士生导师,中国计算机学会(CCF)杰出会员,主要研究领域为嵌入式系统开发、硬件安全、3D 打印、无线网络.E-mail:qwang@xidian.edu.cn.**杨鹏飞**,博士,讲师,中国计算机学会(CCF)会员,主要研究方向为嵌入式系统架构与安全.

research, novel hardware Trojan design and defense methods continue to emerge, thus making the hardware Trojan and its associated techniques also present some new features. In order to understand the hardware Trojan problems more comprehensively and illustrate the latest research progress and development trend in recent years, it is necessary to re-sort out its technical characteristics, research status, and development trends. In this article, we first present a brief description of the basic concept, composition, attack mechanism, classification, and other related techniques in hardware Trojan, and summarizes current research achievements and the latest advances at home and abroad. It then discusses the key problems for the research of hardware Trojan, concluding seven types of research hotspots, i. e., hardware Trojan design, Trojan detection techniques, design-for-trust mechanisms, runtime defense, component level protection, architecture level protection, evaluation and analysis. After that, this article makes a detailed summary and comparison of the research progress for each type of the research hotspots, and makes a comment on the current research works and the problems, application stages, tools, working principles, applicable scenarios, and characteristics existing in the current study respectively. Finally, this paper also discusses the development trend for hardware Trojan research and the relevant techniques in the future, which would be helpful to clearly demonstrate the challenges faced in current research work, promote the research and development of hardware Trojan detection and corresponding security protection technology, and provide a valuable opportunity for researchers those who want to engage in hardware Trojan and its related technology research.

Keywords hardware security; hardware Trojan design; countermeasures; evaluation and analysis

1 引 言

当今时代,信息安全问题面临前所未有的挑战.在半导体供应链全球化趋势下,传统的将底层硬件视为安全的,并以此为基础实施安全防护的策略逐渐失效[1].如图 1 所示[2-3],不可信实体直接或间接参与到电子设备或集成电路(Integrated Circuit, IC)生命周期的各个阶段,从而导致设计、制造、测试、部署和应用各个环节都存在针对硬件的安全漏洞[1-2].攻击者可以利用这些漏洞达到篡改原始设计、降低电路性能、监听控制、拒绝服务、泄露机密信息的目的,甚至可以对 IC 造成不可逆的破坏[3-4].

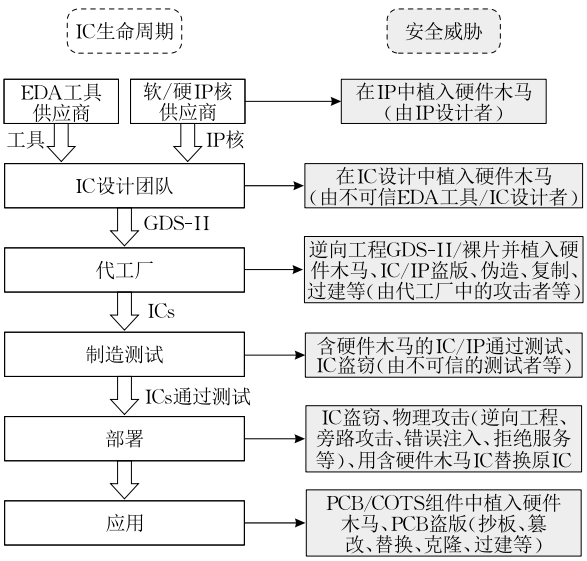


图 1 现代 IC 供应链中的安全威胁[2-3]

IC 安全问题由来已久.早在 2005 年,美国国防部就已经注意到 IC 供应链的安全问题.Adee 在 2008 年的报告中称,叙利亚雷达系统的严重故障可能是通过隐藏在一个商用现货供应(Commercial Off-the-Shelf, COTS)微处理器内的“硬件后门”① 故意触发的[5].2012 年伊朗布什尔核电站在信息系统物理隔绝的情况下遭到“震网病毒”的攻击也是通过后门电路触发产生的[6].据中国台湾媒体报道,中国台湾威盛电子(VIA)公司于 2014 年 11 月 24 日在中国香港高等法院审理“HKIAC/A11022 仲裁案”上诉案中,承认所生产的 VT3421 安全芯片中留有后门②.中科网威公司宣称,确认 Intel 公司的 X86 芯片存在功能不明确的“多余”模块[7].美国国

① 本文如无特殊说明,“硬件木马”与“硬件后门”、“安全漏洞”等概念含义相同或相近.

② Hua Z. Mobile phone “back door” incident, how can via self-justified? http://tech.hqew.com/fangan_1973251, 2017,10,19

防科学委员会于 2017 年 3 月发布报告称，美军武器系统可能已经被注入“硬件后门”，一旦进入实战，敌方完全能够令美方武器陷入瘫痪①. 2018 年 1 月，Google Project One 团队宣称 Intel 公司的 CPU 芯片存在由设计缺陷导致的 Meltdown 和 Spectre 两组“安全漏洞”. 国外媒体 TechSpot 于 2018 年 6 月 27 日报道称，阿姆斯特丹 Vrije 大学的研究人员再次发现 Intel 处理器存在新的超线程漏洞 TBLeed②. 另根据一位不愿透露姓名的美国防务承包商的说法，一家“欧洲芯片制造商”最近为了这种目的设计并制造了带有远程终端开关的微处理器[1].

现代 IC 供应链异常脆弱，所谓的“硬件安全”问题在过去的十几年里受到学术界、工业界和政府的高度关注. “硬件木马”作为其中最为突出的一个安全威胁，已经成为信息与硬件安全领域中一个亟待解决的问题，引起了学术界和工业界越来越广泛的关注.

国内外已有对硬件木马及相关技术的综述类研究[3-4,6-16]. 例如，Bhasin 等人[3,8]对硬件木马的攻击威胁与检测技术作了介绍，主要讨论了逻辑测试和旁路分析技术. Karri 等人[9]在 Tehranipoor 等人[6]调研工作基础上对硬件木马的设计与分类作了进一步的总结与归纳，但没有结合木马实例详细介绍. Zhao 等人[7,10]从检测方法和安全性设计两个方面阐述了硬件木马防御技术，主要探讨了制造阶段的木马检测方法. Jacob 等人[4,11]从 IC 设计流程角度对硬件木马设计、植入难度及相应防护措施进行了系统的描述，主要面向 IC 的功能和物理设计阶段. Forte 等人[1]简要概述了硬件木马近几年的研究情况和发展趋势，而赵剑锋等人[7]则对硬件木马概念的发展过程作了梳理. 上述典型工作主要针对制造阶段硬件木马的相关研究情况进行了调研，重点对旁路信息分析或非破坏性侵入技术进行归纳与探讨，而参考文献发表时间大都集中于 2014 年以前. 随着研究的深入，新的硬件木马设计与防御技术也在不断涌现，有必要对其进行重新梳理归纳.

为了更全面了解“硬件木马”问题，并对近几年最新研究进展进行归纳总结，本文全面梳理近十年国内外关于硬件木马及其相关技术的论文与成果，讨论了硬件木马研究中的关键问题. 针对硬件木马的研究热点内容进行了详细的汇总对比，总结了当前研究工作的成果及存在的不足，探讨了硬件木马及相关技术的发展趋势.

本文的主要贡献如下：

(1) 结合硬件木马危害的真实案例，对硬件木马近十年国内外的研究工作进行了全面梳理，重点展示了近三年所取得的最新研究成果.

(2) 讨论硬件木马研究的关键问题，总结了相关研究热点，并对各研究热点所涉及的研究内容及特点进行了详细的对比分析.

(3) 重新整理了硬件木马检测技术，将其研究范围由制造阶段检测扩展到设计阶段检测，同时将逆向工程由破坏性扩展到非破坏性.

(4) 总结了系统部件级与架构级安全防护技术研究新动向，分析了当前研究工作的成果及存在的不足，并探讨了硬件木马及相关技术的发展趋势.

本文第 2 节对硬件木马及其相关技术研究的关键问题进行阐述，并对国内外的研究概况进行总结；第 3 节对硬件木马设计中关键问题的研究进展进行论述；第 4 节详细分析并讨论硬件木马防御技术中各关键问题的研究进展，突出系统部件级与架构级安全防护技术；第 5 节对硬件木马技术的评估与分析工作进行归纳与整理；第 6 节对硬件木马及相关技术的发展趋势进行探讨；第 7 节对全文进行总结.

2　硬件木马及研究概况简介

2.1　硬件木马的概念

硬件木马是指在 IC 设计或制造过程中被蓄意植入或更改的特殊电路模块、或者是设计者无意留下的设计缺陷[3-4,12]. 当其以某种方式被激活后，可能改变 IC 的功能或规格，泄漏敏感信息，造成 IC 的性能下降、失去控制，甚至是不可逆的破坏[6-8]. 它可以独立完成攻击，也可以与软件协同完成[8]，从几乎不设防的硬件底层潜入，能够直接绕过软件安全防护窥探用户行为，而用户对此却毫不知情. 传统的形式验证和测试工具无法很好检测到这种安全威胁，目前的设计流程也无法保证消除这种安全威胁.

硬件木马不同于制造缺陷. 制造缺陷是无意或随机发生的故障，其行为可以用固定型故障、延迟故障等模型表示[12-13]，而硬件木马是由攻击者专门设计并精心隐藏的，无法用固定模型进行表达[1,11]. 此

① Chen S, Li Z F. Beware of the hidden “hardware Trojan horse” in the area of great security. http://www.sohu.com/a/210403911_262340, 2017,12,14

② Thubron R. Researchers warn of new Hyper-Threading-based Intel CPU vulnerability. http://www.techspot.com/news/75240, 2018,6,26

外，制造缺陷仅在制造过程中产生，而硬件木马可以在 IC 开发的任何阶段插入[11]. 因此，硬件木马问题比制造缺陷表现的更为复杂.

2.2 硬件木马的组成及攻击机理

硬件木马主要由触发逻辑和有效负载两个功能部件组成. 图 2 展示了一种硬件木马的结构模型. 其中，触发逻辑通过监听输入信号、数据/控制总线、寄存器状态、或是经设定的工作时间等方式来激活有效负载；而有效负载则是硬件木马的攻击单元，负责执行攻击行为.

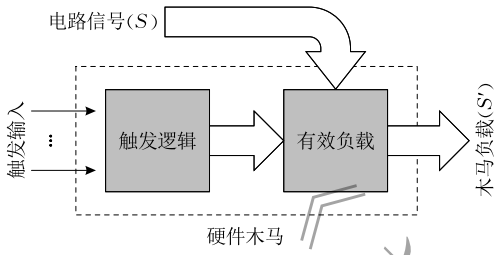
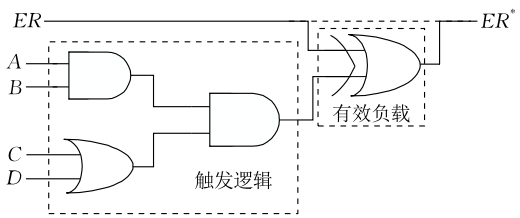
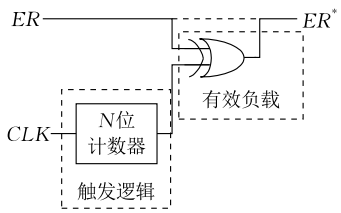


图 2 一种硬件木马结构模型[3]

硬件木马通常处于静止状态，当经特殊条件触发时激活有效负载，通过执行预定的行为，达到攻击者的目的. 图 3 列出了两种常见的硬件木马实例.



(a) 组合硬件木马实例



(b) 时序硬件木马实例

图 3 几种硬件木马实例

其中，图 3(a)是一种简单的组合木马，触发逻辑为一个与门和一个或门，有效负载是一个异或门. 当输入端 A 和 B 同时为 1、C 和 D 至少一个为 1 时触发，通过后面的异或门将 ER 的值改变为 ER^*. 而 A、B、C 和 D 在其它输入情况下，输出结果不改变. 图 3(b)则是一种通过 N 位计数器触发的时序木马. 木马电路通过监测时钟信号 CLK 实施攻击，当计数值达到 2^N-1 时，硬件木马触发，从而改变了 ER 的值为 ER^*.

2.3 硬件木马分类

硬件木马的分类方法有多种，可根据其不同特性，从不同角度进行划分[3,9,12,14-16]. Tehranipoor 等人[12]和 Sumathi 等人[13]提出了基于属性的分类方式，根据插入阶段、抽象级别、激活机理、影响和位置五种不同的属性来对硬件木马进行描述. Moein 等人[15-16]对 Karri 等人[9,12]的分类方式作了进一步的整合与扩展，将五种不同的分类属性扩展为八种，添加逻辑类型、物理布局以及功能特性这三种属性来完善硬件木马分类. 如图 4 所示，这种分类方式能够为硬件木马建立更为详尽的分类模型，并将更多木马类型涵盖在内，方便硬件木马后续的研究工作展开.

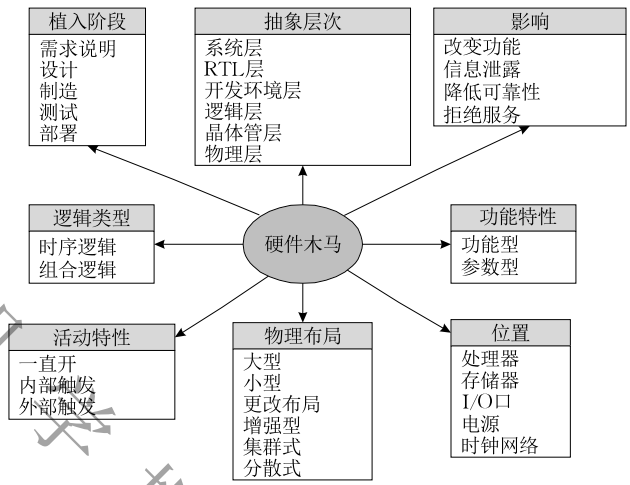


图 4 硬件木马分类[15-16]

2.4 硬件木马研究的关键问题

硬件木马及相关技术研究主要涉及硬件木马的调查研究、攻击策略和安全防御策略三个方面的内容，每个方面又包含若干个关键问题，如图 5 所示.

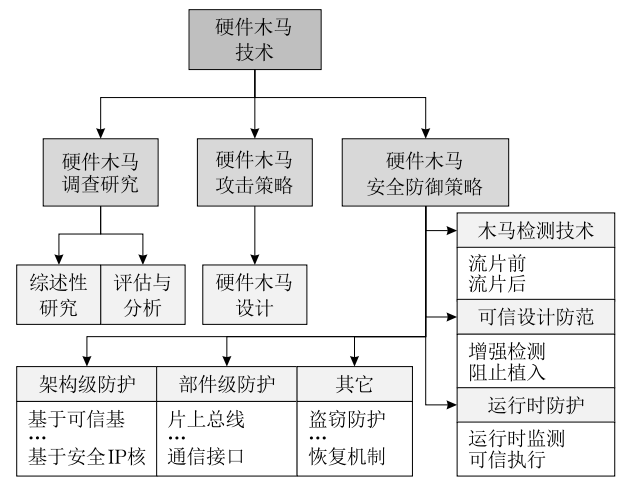


图 5 硬件木马技术研究分类

(1) 硬件木马设计

为更好的防御各种硬件木马，首先要研究其攻击策略. 而对于硬件木马设计技术的研究，将有助于研究人员充分了解硬件木马的植入阶段、物理特性、触发条件、作用机理以及对 IC 的影响等.

(2) 硬件木马防御技术

硬件木马防御技术中所研究的关键问题可归纳为：

① 木马检测技术. 发现在 IC 开发流程的任意阶段中恶意植入到 IC 设计内部的微小电路；

② 可信设计防范. 采用某种辅助手段或防范措施来进一步增强 IC 设计的安全性，达到增强验证、阻止硬件木马植入或触发的目的；

③ 运行时防护. 对逃脱了制造后检测并潜伏进硬件系统中的硬件木马进行在线检测与监测，或是在不可信部件或知识产权(Intellectual Property，IP)核存在情况下执行可信操作；

④ 部件级防护. 对存在于硬件多核系统的片上总线或通信接口等部件中的、并可能引发错误注入、拒绝服务等攻击的硬件木马进行研究，保障片上通信的安全性；

⑤ 架构级防护. 侧重于应对硬件多核系统体系架构层上的硬件木马攻击，尤其是针对片上系统(System-on-Chip，SoC)等嵌入式微处理器；

⑥ 其它. 其它的关键问题还包括针对 IC 与 IP 核的盗窃防护、三维(Three-Dimensional，3D) IC 的安全性以及恢复机制研究等.

(3) 硬件木马技术评估分析

主要针对不同的硬件木马攻击与安全防护技术进行性能分析与比较，常用的评估指标有：敏感度、错误率、时间、成本等.

2.5　国内外研究概况

最早有关硬件木马的论文是由 Agrawal 等人在 2007 年公开发表[17]，截至 2017 年底，国内外在 IEEE①、万方②、知网③等检索上可查到的有关硬件木马研究的期刊、会议论文约有 800 余篇(不包括硕、博论文以及专利等). 每年发表的硬件木马相关论文数如图 6 所示.

由图 6 可见：从 2007 年—2014 年，国内外每年发表的相关论文数呈递增趋势；从 2014 年—2017 年底，硬件木马技术的研究逐渐趋于平稳. 国外关于硬件木马的研究起步较早，而国内关于硬件木马的研究起步较晚，至 2010 年才出现相关的中文文献，但近几年也逐渐成为我国学术领域的一个重要研究方向.

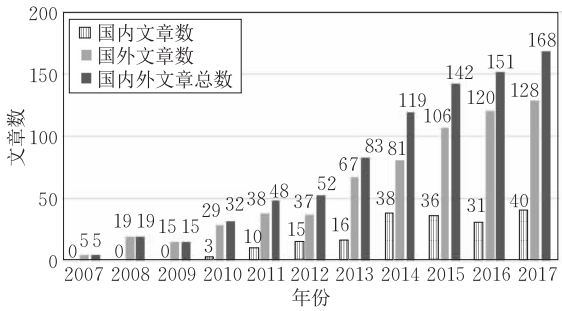


图 6　每年发表的硬件木马相关论文数

有关硬件木马关键问题研究公开发表的学术论文所占比例如图 7 所示.

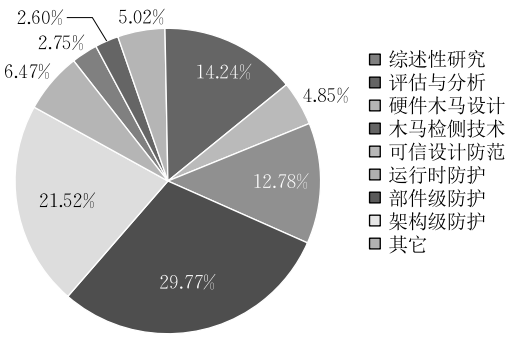


图 7　硬件木马各关键问题研究领域发表论文所占比例

其中，木马检测技术和可信设计防范是两个重要的研究方向，其论文数分别占硬件木马相关总论文数的 29.77%和 21.52%. 此外，对于硬件木马综述性研究、硬件木马设计等研究方向，论文所占比例依次是 14.24%、12.78%. 其它方面的研究涉及运行时防护、评估与分析、部件级防护、架构级防护等问题，论文所占比例依次为 6.47%、4.85%、2.75%、2.60%. 而对于其它问题如 IC/IP 盗窃防护、3DIC 的安全性、恢复机制等问题的研究相对较少，所占总比例仅为 5.02%.

国家自然科学基金自 2011 年开始对硬件木马相关项目进行资助，至今共资助 10 项相关项目. 其中，具有代表性的资助项目有中国人民解放军军械工程学院李雄伟、张阳等人所负责的“基于旁路分析的硬件木马检测关键技术研究”[18-19]；天津大学赵毅强所负责的“无参考模型的硬件木马检测技术研究”[20-21]以及南京航空航天大学薛明富所负责的“免于参考芯片的指纹自认证硬件木马检测方法研究”[22-23]等. 上述工作的研究内容涉及基于功耗等旁路信息的硬件木马离线/在线检测. 应用场景涉及安

① IEEE. http://ieeexplore.ieee.org/Xplore/home.jsp
② WanFang Data. http://www.wanfangdata.com.cn/index.html
③ CNKI. http://www.cnki.net/

全芯片、片上网络芯片、微处理器等多个方面，与国内在这方面的研究情况基本一致.但是，目前国内的大部分研究工作都是对国外已有的研究工作的改进，仍需作进一步的探索与研究.

3 硬件木马设计关键问题研究进展

为了更好的抵御硬件木马攻击，首先我们需要熟悉其设计原理与实现过程.现有的硬件木马设计研究可以分为4类，如图8所示.

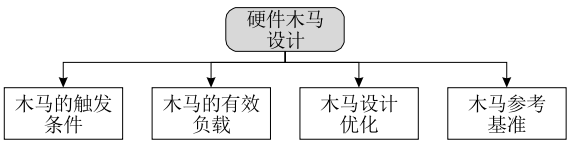


图8 硬件木马设计[1]

(1)触发条件

硬件木马可设计为条件触发和常开两种类型.常开型木马仅包含有效负载电路，木马时刻处于激活状态且极易被发现，故常用于实验验证[9,14].而条件触发型木马包含触发逻辑和有效负载两部分，触发条件既可能是逻辑信号[15-16]，如将电路内部的不相关状态[24]或稀有节点[25-28]等作为硬件木马的触发条件；也可能是模拟信号，如利用电荷[29]、温度[30-31]、或者硅损耗机制[32-33]等电路参数来激活硬件木马.

(2)有效负载

攻击者可能会设计改变电路功能特性的木马电路，如改变电路内部节点的状态或信号值[29-31]等.此时，木马的行为方式是显性的，使电路直接表现出错误行为.木马也可能会利用旁路信息[34-35]或隐蔽通信[36-37]等方式泄露机密信息、或是增加硬件的访问权限[38]、降低电路性能[39]、甚至直接造成物理性摧毁[40].此时，木马的行为方式是隐性的，它不会直接改变电路内部逻辑，而是通过其它方式来实施攻击.

(3)设计优化

硬件木马的植入可能会引起额外的IC开销，例如硅面积、功耗、延时等，这一特性可以用来检测硬件木马的存在[1,3,6-8].为了增加硬件木马的隐蔽性，降低其被检测出来的可能性，研究人员提出对木马电路进行优化，尽可能减少其对原始设计的影响[38,41].

(4)参考基准

学术界还针对不同阶段(RTL、Gate、Layout)和类型的硬件木马提出了一系列的参考基准用以对各种检测方法进行公平、有效的评估[42-45].例如，Trust-Hub.org的基于高级加密标准(Advanced Encryption Standard，AES)单元、RS232等的硬件木马参考基准电路①.

理论上，硬件木马可以在IC设计流程的任何阶段植入.考虑到硬件木马植入所需的成本、时间开销等因素，攻击者在不同阶段植入硬件木马的难度是不同的[7-8,11].表1对本文中所涉及的当前硬件木马设计的部分已有的研究工作进行了汇总.

表1 硬件木马设计研究进展

文章	植入阶段	攻击行为	外部触发	内部触发	
				常开	条件
Fern[24]	功能设计	信息泄露			X状态
Hasan[31]	功能设计	改变功能			温度
Hu[34]	功能设计	拒绝服务			稀有条件
Chang[35]	功能设计	信息泄露		●	
Cheng[37]	功能设计	信息泄露			稀有条件
Koppe[39]	功能设计	降低性能		●	
Patil[30]	物理设计	改变功能	温度		
Li[40]	物理设计	永久破坏	射频		
Yang[29]	制造	改变功能			电荷

注：功能设计指代码开发阶段；物理设计指综合、布局布线阶段.

从表1中可以看出，硬件木马的设计/植入大都集中在设计与制造阶段.其中，以功能设计阶段居多.在该阶段植入硬件木马所需的时间和成本开销最低，难度也最小.物理设计阶段植入硬件木马的时间和成本开销相比于功能设计阶段会有所增加，并且需要攻击者熟知IC的功能设计和电子设计自动化(Electronic Design Automation Tool，EDA)工具等，相应的植入难度也会有所增加.而在制造阶段植入硬件木马的难度最大，所需时间和成本开销也最高.因攻击者对设计了解有限，故存在的风险也最高.因此，硬件木马的设计/植入难度高度依赖于攻击者的时间与成本开销、技术背景、对IC原始设计的熟知程度以及木马电路的植入阶段等.

4 硬件木马防御关键问题研究进展

4.1 木马检测技术

硬件木马检测是处理由硬件木马所引起的安全威胁的最直接、最常用的方式，它旨在验证所设计或制造的IC是否“感染”了硬件木马[6-9,11,15-16].图9给出了当前主要的硬件木马检测方法分类.硬件木马检测主要在两个阶段进行，分别是设计阶段(流片前)检测和制造阶段(流片后)检测.

① Trust-Hub. https://www.trust-hub.org/benchmarks.php

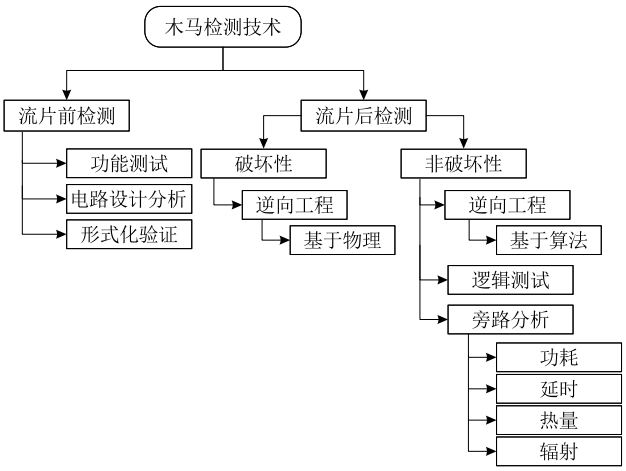


图 9　硬件木马检测方法分类[1,3]

4.1.1　流片前检测

流片前检测主要是对设计模块/单元、或者是所使用的第三方(Third-Party，3P)IP 核进行一系列可信验证，保证 IC 最终设计的安全性[1,11]. 流片前检测技术主要包括功能验证、电路设计分析、形式化验证等.

(1)功能验证

功能验证是指对待测 IC/IP 施加测试激励，并将响应结果与真值表进行比较[11]. 总的来说，功能验证更倾向于发现设计时的功能缺陷，保证所设计 IC/IP 的可靠性. 现有的功能验证技术也适用于对硬件木马进行检测，用以确定设计阶段是否有硬件木马植入. 例如，Sullivan 等人[46]提出对 IP 核进行功能识别，以此来发现是否存在硬件木马. 但是，该方法仅仅是对 RTL 级 IP 核进行简单的信任评估，即验证 IP 核是否满足功能设计所要求的“不多不少”原则[4]. 而对于那些不增加、删除或修改原始设计功能的木马实例，该方法有些力所不及.

(2)电路设计分析

电路设计分析技术是指通过对所设计 IC 的行为[45]或结构[47]代码进行分析，识别可疑的语句或模块，并利用定量指标对疑似为硬件木马的信号或门电路进行标记[48-50]. 例如，Qiu 等人[51]提出对电路属性进行覆盖率分析来发现设计阶段由 EDA 工具恶意植入的代码. 但这一过程非常耗时，可能需要几个小时甚至是几天时间来完成验证过程，并且该方法的扩展性较差. 而 Oya 和 Haider 等人[52-53]则尝试从木马参考基准中提取某一类硬件木马的共性特征，以此来实现对 IC 的漏洞分析. 但该方法在分析较大电路时效果较差，且时间复杂度为多项式级，方法的执行时间和可靠性依赖于电路内网线数以及所选取的定量指标.

电路设计分析是一种典型的启发式检测方法，并且在实践中非常有效. 但它的局限性在于：这一过程非常耗时，即使待测电路的行为或结构属性 100%通过了覆盖率分析，也仍无法保证完全检测出硬件木马，仍需要手动的后期处理来进一步分析可疑信号或门电路，以确定它们是否为硬件木马(或是木马电路的一部分).

(3)形式化验证

形式化验证技术是一种基于算法的逻辑验证方法，它能够详尽地证明 IC 设计应满足的一系列预定义的安全属性规则[54-55]. 为验证 IC 设计是否符合既定属性的要求，可将目标设计转换为 Coq. 审校格式[56-63]，如 Farahmandi 等人[56]提出将 IP 核的门级网表转换成代数多项式并进行形式化验证，用于发现目标电路中的非预期行为；或是利用目标 IP 的行为级特征进行安全属性校验[57-58]，如 Portillo 和 Vedula 等人分别提出利用 IP 核的有效属性[59]或未授权的信息泄露[60]等行为进行安全验证. 然而，Coq. 审校格式转换方法的成本太高，且需要“黄金设计”作比对；而行为特征验证方法主要用于检测信息泄露型木马，扩展性较差. 此外，形式化验证技术可能无法检测到由硬件木马引入的满足既定属性要求的非预期的额外功能，这也是形式化验证技术所存在的不足[61-63].

表 2 对流片前硬件木马主要检测技术特点进行

表 2　设计阶段主要检测技术特点对比

技术名称	适用阶段	借助工具	工作原理	适用场景	特点	存在问题
功能验证	功能设计	仿真软件	施加测试激励，将响应与真值表进行比对.	设计模块/单元/3PIP	操作简单、易于实现、能够发现设计时的功能缺陷.	测试过于简单，测试向量多，测试周期长，应用范围有限.
电路代码/结构分析	功能设计	仿真软件	条件/语句/结构/分支/节点覆盖率分析	行为/结构代码	识别疑似为硬件木马的冗余语句或电路，典型的启发式方法，在实践中非常有效.	仅仅分析设计的组合部分，可靠性无法保证，无法检测未知类型的木马，需进一步手动分析且非常耗时.
形式化验证	功能设计/物理设计	仿真软件/逻辑验证算法	将目标设计转换为 Coq.审校格式，进行属性验证.	RTL 设计/门级网表/GDSII 网表	使用数学推理等方式来验证 RTL 设计与门级网表、门级网表与 GDSII 网表是否一致.	非直接检测木马，而是进行可信评估. 无法检测满足既定属性的木马，扩展性差且对大型设计效果不明显. 无法自动实现 VHDL/Verilog 代码到 Coq.审校格式转换.

对比.从表 2 中可以看出,设计阶段检测主要借助 EDA 仿真工具和逻辑验证算法对 IC 设计进行可信评估,能够帮助识别出设计模块/单元/3PIP 中的可疑部分,高度可信地认定这些可疑部分是否是木马电路(或是木马电路的一部分).但是,上述检测手段仅仅是对待验证 IC 设计进行信任评估,无法保证可靠地检测出在设计阶段植入的所有类型木马,并且验证过程耗时严重,仍需要做进一步检测.

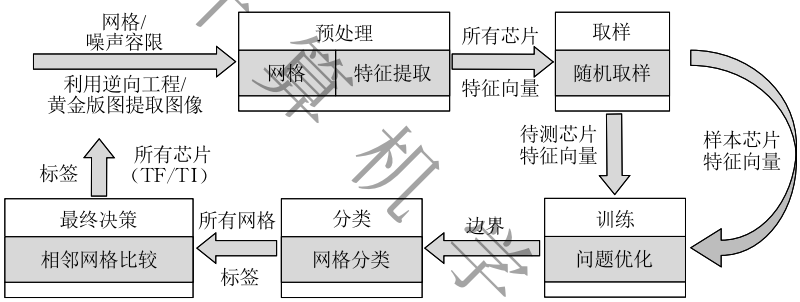
4.1.2 流片后检测

流片后检测主要是对由第三方代工厂生产的、或是部署后的 IC/IP 等进行硬件木马检测.该阶段可分为破坏性和非破坏性两类检测方法,

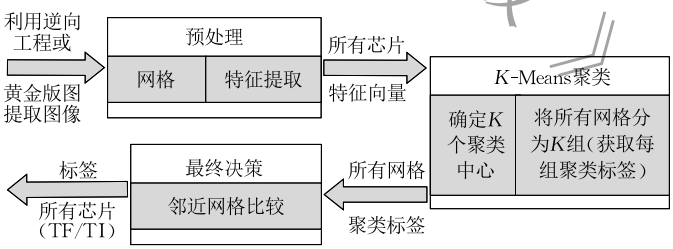
(1)破坏性检测方法

破坏性检测方法通常利用破坏性的逆向工程来对 IC 进行解封装并获得每层的图像,以重现最终产品的可信设计验证[1,3,64-66].逆向工程是不可逆的,在侵入过程结束后 IC 将不能再使用,我们只能得到单个 IC 样品的信息[6-8,11].

然而,为了获得大批量 IC 的电路特性,在有限数量的样品上进行破坏性逆向工程仍具有一定的吸引力[1,3,7].例如,Courbon 等人[64]提出一种快速侵入式技术,该技术首先将待测 IC 进行区域划分,然后利用图像扫描等方式提取每个分区的图像,最后进行多图像拼接以重现原始设计.这一侵入式方法虽然仅对制造阶段通过修改功能单元而植入的硬件木马有效,但却降低了逆向过程的时间和成本开销.Bao 等人[65-66]提出利用支持向量机(Support Vector Machine,SVM)和(K-means)聚类等机器学习方法来识别无硬件木马的 IC.具体操作过程如图 10 所示,首先利用逆向工程/版图对待测 IC 进行图像特征提取、采样,然后对样本作训练与优化操作,最后利用机器学习进行分类.该方法缩减了侵入操作的步骤,避免了手动输入门级网表操作,但在最后的分类与决策阶段仍需要与"黄金 IC"作比对.



(a)基于SVM的硬件木马检测[65]



(b)基于K-means聚类的硬件木马检测[66]

图 10 基于机器学习的硬件木马检测方法

(2)非破坏性检测方法

非破坏性检测方法通常利用非侵入式手段来检测 IC 中的硬件木马.其中,非破坏性逆向工程通过对 IC/IP 设计的门级网表、GDSII 网表、裸片等进行逆向操作来识别疑似木马电路,而逻辑测试和旁路信号分析则用于验证不可信制造厂商所生产的 IC/IP[1,3,6].

①非破坏性逆向工程.

近几年,对于逆向工程技术的研究已经不再局限于破坏性的物理级 IC 逆向,学术界和工业界还提出了非破坏性的算法级逆向工程技术,用以实现从门级网表/GDSII 网表/裸片等到硬件高级描述的逆向[67-68].例如,Li 等人[69]提出利用行为模式挖掘将未知 IC 逆向成输入-输出轨迹模型,并利用行为模式匹配方法来发现木马电路.但是这一过程仅仅勾勒出了原始设计的抽象模型,并不能完全代表其真实的功能,而对于需要特殊序列触发的时序型木马,无法做到很好地发现它们,需要进一步研究完善.

② 逻辑测试

逻辑测试方法的主要思想与之前描述的功能验证相类似，都是通过施加功能/结构/随机测试向量来激活硬件木马，并将响应与正确的结果进行比较[1,6]. 不同的是，功能验证是通过仿真来实现，而逻辑测试则是在专用测试仪器或平台上进行，方便测试模式的输入以及响应输出的采集. 然而，攻击者可以设计稀有触发条件，以便在制造测试过程中逃过传统的功能和结构测试. 为此，研究人员尝试利用新的测试模式生成方法来提高 IC 内部低翻转率节点的活性及木马对主输出影响的可能性[26]. 例如，Kamhoua 和 Saad 等人[27-28]提出利用博弈论确定最佳的测试集，从而提高数字 IC 中硬件木马的检测概率. 而 Voyiatzis 和 Kitsos 等人[70-71]则分别提出利用组合测试原理来减少测试向量数.

然而，IC 内部有许多状态节点和门电路，将其全部列举出来不切合实际. 此外，有些木马并不篡改原始电路的数据或功能，而是通过天线等泄露机密信息，或者只是单纯的修改设计规范，基于逻辑测试的方法无法检测出这类木马[14].

③ 旁路分析

旁路分析方法通常借助 IC 正常工作时的电路参数，例如延时[72]、功耗[17,73]、热辐射[74]、电磁[75-76]等，来检测硬件木马的存在. 它充分利用了由额外电路和/或硬件木马的活动特性所引起的旁路信息改变这一特性，弥补了逻辑测试方法的不足[1,3,8,11].

各大高校和研究所在近几年也纷纷开始研究利用机器学习与旁路分析相结合的方式来提高硬件木马检测的准确率. 通过将木马检测问题重新描述，然后利用机器学习方法实现对待测样本IC的分类操作[77-78]. 例如，Li 等人[79-80]提出利用反向传播(Back Propagation，BP)神经网络分别对待测 IC 的噪声和功耗信息进行提取与分类. 该方法首先将噪声和功耗等旁路信息作为 IC 特征进行提取，然后与参考模型进行匹配，最后通过模式匹配等方法进行分类处理，具体操作过程如图 11 所示.

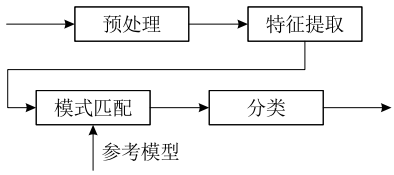


图 11　基于 BP 神经网络的硬件木马检测方法[79-80]

机器学习技术与旁路分析方法的结合能够提高硬件木马检测的分辨率和灵敏度，但采样信息和分类算法的选择是这一应用所面临的重要挑战. 另一方面，旁路信息分析技术大都假定有可用于比较的“黄金设计/IC”，而在实际应用中往往难以获得可用于比较的“黄金设计/IC”，这是此类方法所面临的另一个挑战. 另外，旁路分析方法虽然能够在一定程度上检测到木马电路，但是在过程变化和环境噪声存在的情况下，实现较高的覆盖率以及提取由木马电路所引起的旁路信息的微小差异比较困难.

表 3 对流片后硬件木马主要检测技术特点进行对比. 从表 3 中可以看出，制造阶段检需要借助于专用测试仪器/平台来完成整个测试过程. 其中，逆向工程虽能够对 IC 进行彻底检测，发现并识别任何恶意的修改，但该方法耗时长且只能通过抽样方式进行，不适合对 IC 进行逐一排查；逻辑测试不受噪声影响且对小规模木马有效，易于自动化实现，但测试向量生成困难，不适合检测大规模木马；而旁路分析

表 3　制造阶段主要检测技术特点对比

技术名称	适用阶段	借助工具	工作原理	适用场景	特点	存在问题
物理级逆向工程	制造/测试/部署	光学/电子扫描显微镜、成像电路分析仪、电压对比成像器等	去封装、逐层扫描、重建原始设计、与“黄金IC/设计”比较、硬件木马检测.	制造或部署后的简单逻辑 IC	破坏性检测方法，过程不可逆，检测精度理论上可达 100%.	成本高、过程耗时，会损坏电路，对复杂 IC 效果不明显，设备精度要求高，且无“黄金设计/IC”作比对.
算法级逆向工程	功能设计/物理设计	软件仿真、数学算法	将门级网表还原成高级语言描述或输入-输出模型.	简单逻辑 IC/IP 的门级/GDSII 网表/裸片	非破坏性检测方法、检测准确度高，且不会损坏电路.	成本高、过程耗时，且无“黄金设计”进行比对，仅仅是对原始设计的抽象描述.
逻辑测试	制造/测试/部署/运行	专用测试仪器/平台	施加功能/结构/随机测试向量，识别硬件木马对主输出响应的影响.	特定引脚触发的制造或部署后IC/IP	过程稳定，受噪声影响小，能有效检测出较小的显式木马.	测试模式生成复杂，对较大的显式或隐式木马效果不理想.
旁路信息分析	制造/测试/部署/运行	高精度示波器/温度检测器/功耗分析仪/频谱分析仪等	利用硬件木马对电路参数的影响. 硬件木马激活后会改变电路的电流、电压、温度、延时等.	偏向于制造或部署后IC/IP	检测精度高、条件限制少，不要求木马电路被触发，只要电路部分工作即可. 应用比较广泛，适合检测较大木马.	容易受工艺变量和各种噪声影响，对较小木马效果不理想. 对设备精度要求高，且需要“黄金 IC”进行比对.

容易生成测试向量,适合检测较大的木马,对原始电路无破坏,但却易受工艺噪声影响.通过比较这些方法的优缺点,能够实现各木马检测技术的优势互补,提升硬件木马检测的精度和效率.

4.2 可信设计防范

可信设计防范(Design-for-Trust,DFT)技术主要用来规划设计阶段的木马问题[7,11].该技术主要可分为2个子类,如图12所示.

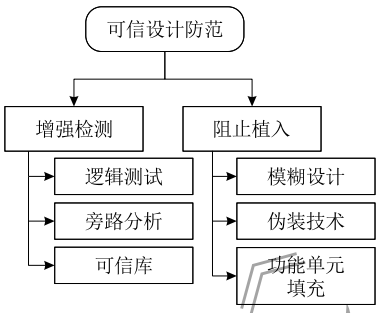


图12 安全性设计方法分类[7]

4.2.1 增强检测

增强检测是可信设计防范的第一类方法,旨在通过增加片上模块等方式来辅助促进非破坏性硬件木马检测方法,放大硬件木马对IC电路参数的影响,或者利用所增加的片上模块实现可信库的构建等.

(1)增强逻辑测试

由于硬件木马固有的隐蔽特性,触发一个木马并观察其输出影响仍十分艰难.主要原因在于IC设计中存在大量低可控性与低可观测性的电路节点,硬件木马可能选择这些电路节点作为触发输入,从而降低了硬件木马有效激活的可能性.

为此,Salmani等人[81]提出通过在IC设计中植入虚拟扫描触发器(Dummy Scan Flip-Flop,DSFF)单元的方式来提高IC内部节点的可控性与可观测性.Zhou等人[82]在Salmani等人[81]的基础上,提出增加2路复用单元(Multiplexer,MUX)来复用DSFF的两个输出端口.这不仅扩大了设计的空间状态,而且增加了将硬件木马影响激发并传播到电路输出端的概率,使得硬件木马更易于检测.

上述方法不仅能够进一步改善逻辑测试技术,提高内部低可控与低观测性电路节点的翻转率,而且能够帮助需要部分激活的硬件木马电路完成基于旁路信息分析的检测,但却会引入额外的硬件或资源开销,甚至可能会降低IC性能.另外,电路内部节点的翻转率与所施加的测试向量以及电路自身的拓扑结构具有一定相关性,这可能会导致所植入DSFF单元的位置及数量差异.

(2)增强旁路信息分析

该方法与旁路分析方法类似,不同的是该方法通过增加测试点或片上模块/传感器等方式来辅助放大木马电路对旁路信息的影响,提高硬件木马检测的灵敏度和可靠性.例如,Wei等人[83-84]提出利用预先植入测试点的方法采集IC内局部时间特征.

此外,一些研究人员还研究利用新型电路结构或传感器以提供比传统测量更高的检测灵敏度.例如,Zarrinchian等人[85]提出利用闩锁电路结构将由硬件木马引起的额外路径延迟转换为可观测的功能变化.Bastos和Ismari等人分别利用片上电流[86]或延时[87]传感器来检测硬件木马对瞬态电流、路径延时等电气参数的影响.

上述方法够在一定程度上放大硬件木马对旁路参数的影响,增强基于旁路信息分析方法的灵敏度和可靠性,但却会增加IC设计的时间与资源开销.另外,作为一种侵入式的辅助手段,所增加的测试点或片上模块/传感器等都可能被攻击者利用,进而造成IC内部敏感信息的泄露等.

(3)可信库

当前的硬件木马检测技术大都假设有可用的"黄金"参考模型作比对,而在实际中往往难以获得,这成为制约当前研究工作实际可行性的重要因素[1,7,10-11].若是能够对某些电路参数信息进行采集并建立可信库模型,则能够在一定程度上消除对"黄金设计/IC"的依赖.例如,对物理级IC进行旁路信息采集,并执行逆向工程操作,可以认为是一种破坏性的可信库建立过程[1,3,64-66].除去4.1.2节中已提到的时间/成本开销等弊端外,该方法还受到不同IC制造厂商与制造工艺/批次的影响.

此外,一些研究人员还尝试在设计阶段对IC进行蒙特卡罗(Monte Carlo,MC)仿真来建立可信库,以此来摆脱对"黄金IC"的需求[88].例如,Xue等人[89]提出在设计阶段对IC进行仿真,并通过自适应迭代优化算法完成对数据集的训练优化.Liu等人[90]提出利用在IC中插入的过程控制监测(Process Control Monitoring,PCM)单元完成对旁路信息的采集与统计,并通过K-means或SVM等机器学习方法来建立可信库,建立过程如图13所示.

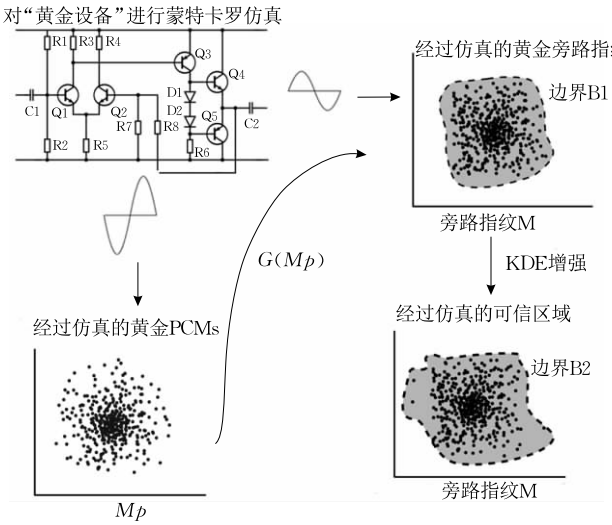


图 13　设计阶段的可信库构建过程[90]

事实上，已有的研究工作仅仅是对设计阶段的 IC 网表进行仿真，所获得的仿真数据往往与真实 IC 的测量结果存在一定的差异. 而采用物理级逆向工程方法时所需“可信”IC 裸片通常难以获得. 因此，可信库的构建仍是当前所面临的一个巨大挑战.

表 4 对增强检测技术中各安全防护方法的特点进行对比. 从表 4 中可以看出，通过在 IC 原始设计中增加额外片上模块/单元/电路结构/传感器等，能够辅助提高非破坏性木马检测方法的敏感度，更好地发挥各自优势，但却会增加电路冗余，对电路性能造成一定影响. 而可信库模型构建时仍需要可信的“黄金”模型作为参考，且所得的仿真实验数据有别于真实 IC 的测量结果，所以仍需要作进一步完善.

表 4　增强检测主要防护技术特点对比

技术名称	适用阶段	借助工具	工作原理	适用场景	特点	存在问题
增强逻辑测试	制造/测试/部署/运行	专用测试仪器/平台	在 IC 原始设计中植入 DSFF 单元等额外的电路结构.	特定引脚触发的制造或部署后 IC/IP	一种侵入式的辅助检测手段，能够移除稀有节点/事件，增加木马激发及其影响输出的概率.	引入额外的面积开销，对性能影响较大. 另外，还需要“黄金 IC”作比对，仅适用于小型设计.
增强旁路分析	制造/测试/部署/运行	高精度示波器/温度检测器/功耗分析仪/频谱分析仪等	在IC原始设计中增加测试点/电路结构/传感器.	偏向于制造或部署后 IC/IP	一种侵入式的辅助检测手段，能够进一步放大硬件木马对电路参数影响，提高检测的灵敏度.	局部监测电路参数. 会增加电路冗余，造成性能降低，甚至信息泄露. 另外，还需要“黄金 IC”作比对.
可信库	功能设计/物理设计/制造/部署	EDA 工具、高精度示波器/温度检测仪/功耗分析仪/频谱分析仪等	MC 仿真/物理级逆向工程/利用在 IC 设计中增加 PCM 单元等.	可信的设计、制造、测试或部署后的 IC/IP	为木马检测技术构建所需的可信参考库，克服对“黄金设计/IC”的实际依赖性.	容易造成额外的面积/功耗开销及信息泄露等，目前仍停留在实验仿真阶段，且需要可信的 IC 设计/裸片.

4.2.2　阻止植入

阻止植入是可信设计防范的另一类方法，该方法试图阻止硬件木马的恶意植入或激活，从而减少木马危害.

(1) 模糊设计

模糊设计试图通过将内置锁定机制(例如，逻辑加密电路等)插入到原始设计中，并且只有在输入正确的密钥时才会执行正确的操作[91]，以此来达到隐藏原始设计真实功能的目的. 这种锁定电路可能是透明的，如果攻击者不知道正确的密钥，识别原始设计的真正功能将会变得非常复杂，从而使插入目标木马的可能性变得很渺小[92]. 图 14 给出了一种基于逻辑加密的模糊设计框图.

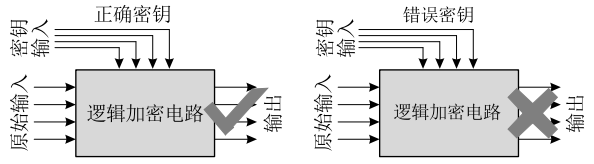


图 14　基于逻辑加密的模糊设计框图[92]

对于基于组合逻辑的模糊设计，可以将复用单元 MUX[93-94]、门电路[95-96](如 XOR、XNOR)、查找表[97-98](Lookup Table，LUT)或物理不可克隆函数[99-100](Physically Unclonable Function，PUF)电路等植入到原始设计中来实现. 对于基于时序逻辑的模糊设计，可以在有限状态机[101](Finite State Machine，FSM)中引入附加状态来隐藏其真实的功能状态. 此外，一些研究人员尝试将可重构逻辑[102-103]植入到原始设计中，只有当这些可重构电路被正确编程时才能执行相应操作.

模糊设计通常在 IC/IP 的功能设计阶段完成，它试图通过混淆电路的功能或结构特性来阻止硬件木马的植入或触发. 尽管该方法十分有效，但却大大增加了 IC 设计的时间、面积与功耗等资源开销，并且可能会对 IC 的性能造成较大的影响. 而对于那些熟知整个 IC 设计的攻击者来说，模糊设计无法做到很好的防御.

(2) 伪装技术

伪装技术是一种布局、布线阶段的模糊设计技

术，通过在设计内部各层之间添加伪装逻辑、或者添加伪接触、伪连接等，来为不同电路创建难以区分的布局[104-105]. 例如，Bi 等人[106]借鉴了类似的虚拟接触伪装方法并研制出一套基于极性可控的硅纳米(SiNW)FETs 与石墨烯 FETs 伪装单元.

伪装技术能够阻止攻击者利用逆向工程等技术从各层的布局图像中提取出正确的门级网表，从而保护原有设计免受木马攻击[107]. 但是，伪装技术会增加 IC 在布局、布线阶段的时间开销，并且所添加的伪接触或伪连接可能引发串扰等影响. 另有研究证明，伪装的电路仍存在被逆向的可能[97].

(3) 布局空间填充

对于 IC 内部那些未使用的空间，设计时通常利用不具有任何功能的空白单元进行填充，其目标是去除可用于插入附加电路的任何空间. 而在布局时插入硬件木马最隐蔽的方式就是替换这些填充的空白单元，因其对电路参数影响最小.

Xiao 等人[108]分别提出了内建自验证(Built-in Self Authentication, BISA)的方法，利用标准功能单元来填充 IC 内部空白区域. 图 15 给出了 BISA 的结构设计. 该技术可将所植入的标准功能单元自动连接成组合测试电路，能够在不影响 IC 功能的前提下阻止对标准功能单元的任意篡改，测试失败意味着木马电路的存在. 但是，BISA 技术所使用的标准功能单元需由专门人员精心设计，且使用量巨大，这将导致额外的硬件资源与成本开销. 为此，研究人员提出了优先填充“关键”空白区域的方法[109-110]. 例如 Hossein-Talaee 等人[111]提出利用空白/拥挤区域重排方法来降低布局高风险区域的漏洞级别，从而减少所使用的功能填充单元的数量，并间接降低硬件木马植入的概率. 然而，布局空间填充方法无法阻止那些仅对晶体管参数进行修改或不增加额外面积开销类型的木马电路.

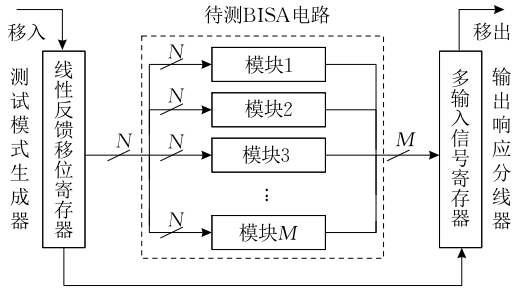


图 15 BISA 的结构设计[108]

表 5 对阻止植入技术中各安全防护方法的特点进行对比. 从表 5 中可以看出，阻止植入技术属于木马植入/触发预防手段，借助于 EDA 工具，开发人员能够在 IC 设计阶段通过设计模糊、伪装、填充功能单元等方式达到迷惑对手的目的，使得对手难以植入/触发木马电路. 此外，该技术还能够有效防止对 IC 设计的逆向攻击，但可能会影响 IC 原始电路的可靠性.

表 5 阻止植入主要防护技术特点对比

技术名称	适用阶段	借助工具	工作原理	适用场景	特点	存在问题
模糊设计	功能设计/物理设计	EDA 工具	在 IC 原始设计中植入内置锁定机制.	RTL 设计/门级网表	侵入式加密防护措施，只有输入正确密钥才会执行正常操作，能够防止木马植入/逆向工程.	造成较大的电路冗余和资源开销，对电路性能影响较大.
伪装技术	物理设计	EDA 工具	在 IC 原始设计内部各层之间添加伪装逻辑、伪接触、或是伪连接等.	门级网表/GDSII 网表	能够迷惑对手，降低硬件木马植入或触发概率，抵抗逆向工程.	对电路性能影响严重，容易引发串扰等影响，无法完全防御逆向工程.
布局空间填充	物理设计	EDA 工具	利用标准功能单元填充 IC 原始设计内部的空白区域.	GDSII 网表	使固有模式可测试其所有门电路，木马预防灵敏度高，标准功能单元鉴别困难.	标准功能单元使用量多，资源与成本开销大，且无法阻止参数型木马.

4.3 运行时防护

虽然已有的相关工作在一定程度上缓解了硬件木马所造成的危害，但仍然具有各自的局限性[3,7-8]. 某些硬件木马可能逃脱了制造后检测并潜伏进硬件系统中. 此时，运行时防护技术成为对抗硬件木马威胁的最后防御手段. 该技术主要包括运行时监测与可信执行两个方面，如图 16 所示.

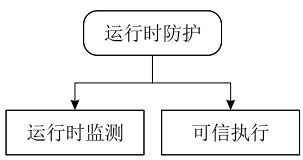


图 16 运行时防护技术分类

4.3.1 运行时监测.

运行时监测方法可以充分利用现有或额外的片

上模块/单元来监测 IC 运行时的行为或状态[112]，如瞬态功耗[113-114]、温度[115]、延时[116-117]等. 此外，研究人员还提出构建片上分类器，并与片上传感器结合，实现对硬件木马的在线检测与分类[118]. 例如，Liu 等人[119]曾提出将 IC 的行为操作与瞬态电流相结合，同时借助于片上神经网络分类器来并发的区分有木马与无木马 IC. 但是，片上神经网络分类器只能对目标 IC 进行局部监测，并且分类器自身的安全性无法保证，也可能会遭受硬件木马攻击.

运行时监测方法尽管会增加电路冗余，但是却能够在检测到任何异常的情况下禁用 IC，或是绕过它来继续执行可靠操作，进而提高对硬件木马的信任等级. 而对于某些任务关键型应用，运行时监测对于硬件木马检测与恢复是十分必要的. 但是该方法只关注于监测点，并不能全面检测整个 IC，对于某些复杂的大型设计，该方法显得有些力不从心.

4.3.2　可信执行

可信执行是运行时防护的第二类方法，它通常利用来自不同供应商的 3PIP 来实现可信验证，比如利用分布式任务调度方式在多核处理器中实现硬件木马激活容忍型的可信计算系统[120-121]；或者是利用投票机制[122-123]与并发错误检测(Concurrent Error Detection，CED)等技术[124-125]来检测硬件木马所产生的恶意输出. 例如，Alkabani 等人[123]试图通过多数表决器来发现可疑 IP 核. 图 17 给出了基于投票机制的硬件木马检测过程，通过在运行阶段对来自不同供应商的功能相同或相似 3PIP 核的输出结果进行比较，发现可能植入木马电路的 3PIP 核[124]. 投票机制与 CED 技术虽能够免除 3PIP 核中硬件木马影响，但却会增加运行阶段的功耗开销，并且会对 IC 的执行效率造成一定的影响.

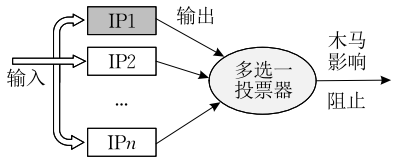


图 17　基于投票机制的硬件木马检测[123]

此外，一些研究人员尝试利用不同 3PIP 供应商之间的分配与约束策略来阻止硬件木马的影响[125-127]. 例如，Sengupta 等人[126-127]利用 3PIP 与不同供应商之间的分配操作以及约束策略来防止来自同一供应商的 3PIP 之间的共谋影响. 但是，该分配操作却无法避免不同供应商之间的共谋影响. 可信执行方法有别于运行时监测方法，原因在于该方法是木马攻击容忍的. 它试图通过部署、调度等方式来保证硬件系统的可信执行. 然而，可信执行可能会导致极高的资源和功耗开销，且无法完全避免 3PIP 之间的共谋影响，因为不同供应商之间可能相互勾结，以此来间接实现这种共谋威胁.

表 6 对运行时防护技术中各安全防护方法的特点进行对比. 从表 6 中可以看出，运行时监测虽能够在线识别并验证 IC 运行时的行为/状态，但只能实现局部监测，无法顾全大局. 而可信执行可以认为是运行时监测方法的更进一步，能够保障硬件系统的可靠运行. 两者之间的最大区别在于是否是木马攻击容忍的，即是否在硬件木马存在的情况下允许 IC 继续执行操作.

表 6　运行时防护主要技术特点对比

技术名称	适用阶段	借助工具	工作原理	适用场景	特点	存在问题
运行时监测	测试/运行	不需要	在IC原始设计中增加额外的片上模块/单元或新型电路结构/传感器.	部署或应用后的IC/IP	一种侵入式的辅助检测手段，能够在线监测 IC 运行时的行为/状态，提高 IC 在运行阶段的安全性.	不能全面监测电路，仅仅关注监测点. 对电路性能及可靠性影响严重，甚至会泄露内部信息，且难以应用于大型设计.
可信执行	部署/运行	不需要	通过布局时 3PIP 之间的约束、任务部署/调度以及投票等多种方式保障多核 IC 的可信执行.	具有不可信 3PIP 核的制造或部署后多核 IC	木马攻击容忍的，避免触发硬件木马，保障 IC 的可靠执行，能够降低 3PIP 共谋影响，侧重于运行阶段.	会影响硬件系统的性能以及执行效率，无法完全避免 3PIP 间的共谋影响.

4.4　系统部件级防护

部件级防护旨在保证硬件系统多核通信之间的安全性与数据传输的可靠性，例如防止多核之间的片上通信总线或通信接口等部件遭受来自外部或内部的安全威胁，例如旁路攻击、错误注入、拒绝服务、硬件木马等[3,6,128].

对于片上通信总线，攻击者可以在不影响正常通信情况下，利用隐蔽通信[129]、总线空闲状态[130]或者通过总线网络接口[131]等方式执行通信数据与路径篡改、非法访问以及信息泄露等操作. 例如，硬件木马可能存在于片上网络(Network-on-Chip，NoC)总线的路由节点中，进行通信链路或数据的篡

改等攻击行为，如图 18 所示.

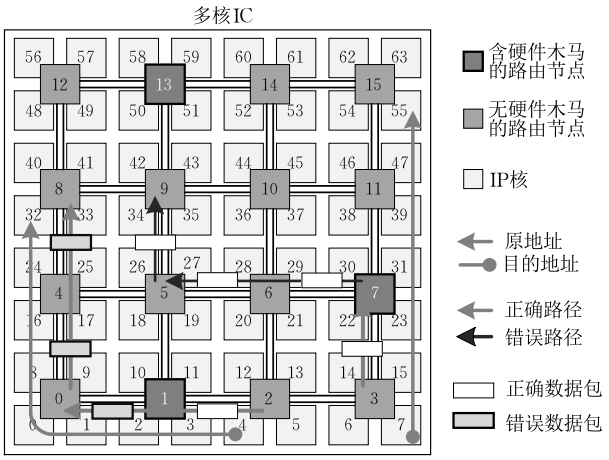


图 18 NoC 总线路由节点中硬件木马的攻击行为

近年来，对于片上通信总线安全方面的研究取得了一定的进展，其中以 NoC 总线的关注最多. 例如，Fery 等人[131] 提出利用状态模糊技术来提高 NoC 网络接口中硬件木马的检测概率. 但是该方法需要密钥输入来创建模糊状态，这不仅增加了面积与功耗开销，而且容易受到错误注入、拒绝服务或旁路攻击等的影响. Boraten 和 Pino 等人分别提出利用机器学习技术，如启发式方法[132-133]、自适应学习算法[134-135]等所构建的安全威胁检测模型来避免 NoC 总线遭受由硬件木马引起的拒绝服务等攻击. 然而，所构建的安全检测模型虽能够增强多核之间数据传输的安全性与可靠性，但构建过程非常耗时，且对模型依赖性较大. 另外，Liu 和 Kim 等人[136-137] 尝试利用总线仲裁与随机数产生相结合的方式来保障存储访问的可靠性.

与可信执行类似，部件级防护也属于木马攻击容忍的范畴，能够有效保证多核 IC 中数据通信与传输的安全性与可靠性. 但是，该方法主要作用在运行阶段，所增加的功能模块会引起额外的面积开销，并可能会对多核间通信的性能、时间以及功耗开销造成一定影响.

4.5 系统架构级防护

现有的硬件木马检测与防护技术对体系架构层面的硬件木马问题涉及较少. 随着现代 IC 集成度的增加以及 3PIP 核的大量使用，设计的复杂度也在不断提高，硬件木马可能会从新的层面威胁硬件系统的安全，研究人员有必要对诸如 SoC 等多核 IC 的架构级安全问题进行研究[138-139].

这方面最早的研究工作是由 Koushanfar 等人在 2011 年提出的统一架构模型[140]，可以看作是体系架构级安全防护研究的雏形. 研究人员随后作了以可信计算基为基础来构建安全体系架构的尝试，如 SAFER PATH 安全处理架构等[141]，旨在保障 COTS 组件中存在活动硬件木马情况下的正确运行. 而另外一些研究人员则尝试构建基于可信评估的安全体系架构模型，以此来增强无线加密 IC[118,142] 或计算机硬件系统[143] 的安全性与信任等级.

然而，上述安全体系架构模型并不适用于异构多核 IC. 为此，学术界和工业界针对 SoC 提出为其构建专有安全体系架构模型的方案[144-149]，以此来增强 SoC 的安全性. 例如，Wang 等人[144] 在 IEEE 1500 嵌入式核测试标准的基础上，提出利用基建 IP (Infrastructure IP，IIP) 核为嵌入式 SoC 构建安全体系架构模型，保障其设计的安全性. 如图 19 所示，该模型利用 IIPS 核与 SoC 内各封装 IP 核进行连接，并将各封装 IP 核连接成扫描链结构，整个测试/验证过程由 IIPS 控制完成.

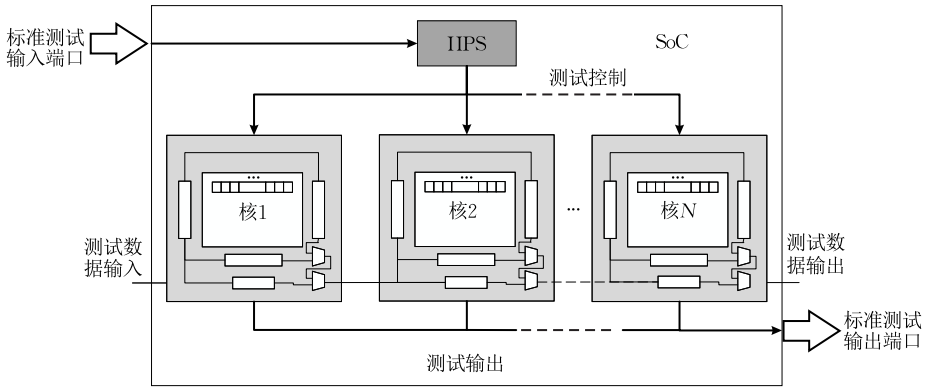


图 19 基于 IIPS 的安全体系架构模型[144]

系统架构级防护与可信设计防范严格来说都属于设计时考虑因素(Design-Time Considerations，DTC)机制.不同的是系统架构级防护优先考虑安全因素，而后考虑功能实现，而可信设计防范仅仅作为一种增强硬件木马检测或防护的辅助手段.

表 7 对硬件多核系统部件级与架构级防护技术的特点进行对比.从表 7 中可以看出，部件级防护侧重于局部，而架构级防护更多的从 IC 设计全流程角度关注硬件系统的安全问题，但是，部件级防护是容忍木马攻击的，而体系架构级防护方法主观性较大，且全面的漏洞分析费时费力.

表 7　硬件多核系统部件级与架构级防护技术特点对比

技术名称	适用阶段	借助工具	工作原理	适用场景	特点	存在问题
多核 IC部件级防护	测试/部署/运行	不需要	保障多核之间的通信与数据传输的不受由硬件木马所引起的拒绝服务、错误注入、功能改变等攻击影响.	多核 IC 的片上通信总线、通信接口以及路由节点等	木马攻击容忍的，针对多核 IC 部件上的木马，实现攻击检测、阻止与故障现场恢复，保障多核之间通信的安全性与可靠性.	无法检测多核 IC 内 IP 核中的硬件木马，检测的准确性对威胁模型依赖较大，无法抵御某些来自外部的安全威胁.
多核 IC 体系架构级防护	功能设计/物理设计/制造/测试/部署/运行	不需要	对整个 IC 的设计流程进行全面漏洞分析，发现潜在威胁，设计相应的片上安全模块/IP，并构建安全体系架构.	IC设计全流程/整个生态链	从系统架构方面入手，优先考虑安全因素，增强 IC 设计流程的安全性，属于主动防御范畴.	漏洞分析与评估依赖于设计团队的经验、资金及所采用的分析与评估策略，主观性大且耗时长.

5　硬件木马技术评估分析

目前，学术界和工业界也开展了对硬件木马技术评估与分析工作，部分研究工作进展如表 8 所示.

表 8　硬件木马评估与分析研究进展

文献	评估目标	基准电路	评价指标	结论
Hely [150]	逻辑测试/旁路信息	ISCAS'85	主输出/敏感度	互为补充，需要“黄金 IC”
Reece[151]	数据泄露型硬件木马	AES 基准电路	面积/功耗开销	木马对电路影响非常小
Ni[152]	测试向量与敏感度关系	AES 基准电路	功率相对变化参数	测试向量影响检测精度
Rithesh[153]	DSFF	ISCAS'89	翻转率	提高翻转率
Kitsos[154]	RO[1]长度	FPGA[2]	敏感度	RO长度影响检测敏感度
Lecomte[155]	寄生参数	AES 电路	RO 频率/内部电压	寄生参数影响检测精度
Kitsos[156]	TERO[3]与 RO 对比	FPGA	时序敏感度	同长度下，两者频率不同

注：1. 环形振荡器(Ring Oscillator，RO)；
2. 现场可编程门阵列(Field Programmable Gate Array，FPGA)；
3. 跃迁效应环形振荡器(Transition Effect Ring Oscillator，TERO).

从表 8 中可以看出：当前的评估与分析研究工作主要对硬件木马检测(如逻辑测试、旁路分析等)与可信设计防范技术(如 DSFF、RO 等)进行实验评估与分析，而对于木马攻击等的评估较少.另外，用于评价各种技术的性能指标也是五花八门，没有统一的标准.从表 8 可以看出，常用的性能评价指标有：敏感度、翻转率、功耗开销、电路参数等.

6　硬件木马发展趋势探讨

在硬件木马检测与防护方面，研究人员虽然已经做了大量工作，且能够在一定程度上降低现代 IC 供应链的脆弱性，但却以点对点防护为主，智能化程度不高，尚未形成系统的防御体系结构.随着硬件木马技术的迅速发展，新型硬件木马不断涌现，也为硬件木马防御技术带来了新的问题与挑战，因此迫切需要对 IC 设计的全流程进行彻底而有效的分析，以提供全面的系统级安全防护.

未来需要研究人员从系统层面、IC 设计全流程的不同层次上全面考察、分析电子设备与硬件 IC 的安全问题.图 20 提出了一种系统级硬件木马检测与防护模型，智能化检测、防护与对抗各种硬件木马攻击.首先对 IC 设计的各个阶段采取特定的方法和措施来防止硬件木马的植入，提高检测概率，实现点对点防护.然后将各阶段、各种防护策略相互融合，构建系统级智能化防御模型，并借助云端大数据来实现实时检测、异常监测与动态预警等.

除上述主要内容外，还有一些问题需要重视，对于这些问题的研究将使得硬件木马技术在未来的发展中呈现出一些新的特点和趋势.

6.1　无“黄金”参考模型的硬件木马检测

现有的硬件木马检测方法，比如逻辑测试与旁

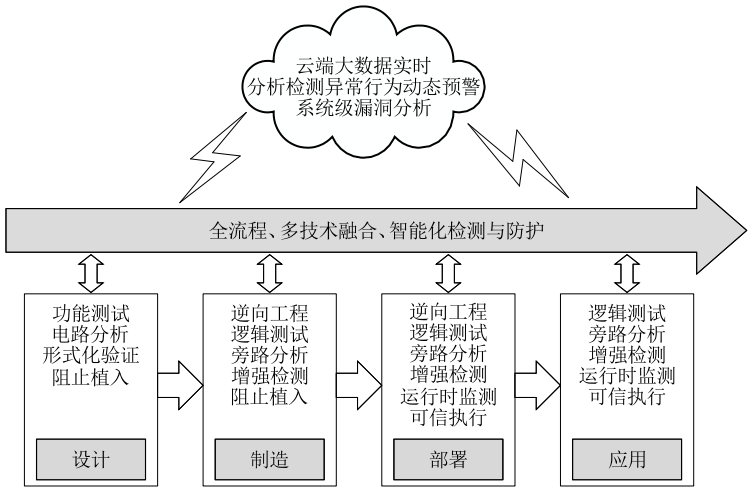


图 20 系统级硬件木马检测与防护模型

路分析等,大都需要“黄金设计”或“黄金 IC”作为参考模型.但在大多数情况下,有一个可用的“黄金设计”是非常不现实的,而单独制造的 IC 难以被用作逆向或旁路检测的“黄金 IC”[4,8,10].因此,研究人员需要开发无“黄金”参考模型的硬件木马检测技术.

Priya 等人[157]提出了一种时间自参考方法,通过比较相同测试模式下不同时间窗口的芯片签名,以完全消除对“黄金 IC”的依赖.但是这种技术只适用于在 IC 中具有不同状态的连续木马检测,并且需要在测试期间保持测试模式的一致.另外,在 4.2.1 节中曾提到过对于可信库构建[33,90].这些技术可以通过建模来消除对“黄金设计/IC”的需求,但其有效性高度依赖于模型的准确性,影响了检测的置信度.

因此,研究人员仍需深入研究无“黄金”参考模型的硬件木马检测技术和方法.

6.2 尝试多种安全防御技术融合策略

硬件木马设计灵活、植入方式与阶段多种多样,而每一种安全防御技术又各具特点,单纯依靠某种防御技术无法很好的抵御其攻击,因此可以尝试采用多种防御技术相组合的策略来提高防御效果.例如,可将逻辑测试与旁路分析相结合,既能够弥补逻辑测试不能检测小型或隐式木马的不足,又能够克服旁路分析易受工艺变量和环境噪声影响的缺点.Koushanfar 等人[140]提出的统一架构模型,尝试将三种检测方法结合来提高硬件木马检测的可靠性.另外,也可尝试将可信执行、运行时监测、部件级防护等安全措施相结合,来提高硬件多核系统运行时的安全性与可靠性.

6.3 硬件木马容忍设计

由于在 IC/IP 设计中很难完全检测并阻止硬件木马,硬件木马容忍设计方法成为保护 IC/IP 设计免受木马攻击的另一种方法.

硬件木马容忍设计主要采用三种策略:第一种方式是尝试消除木马的行为.例如,4.3 节中的运行时防护方法曾提出利用容错设计或结构,能够在硬件木马存在的情况下阻止其恶意影响.另一种策略是阻止硬件木马的触发.由于大多数硬件木马都是由条件触发,这同样也为我们提供了一个机会,即可以通过避免触发这些木马,在硬件平台上实现安全可靠与可信操作.最后一种方法是阻止硬件木马植入.4.2.2 节中已经提到,这里不做赘述.

6.4 恢复机制

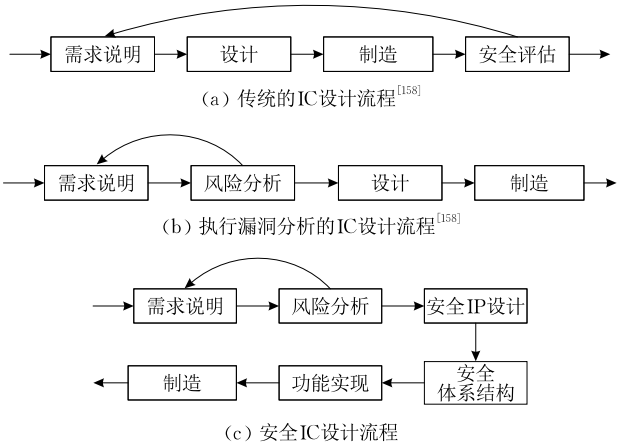
硬件木马成功植入并触发后的故障现场恢复,也是需要考虑的一个问题.

硬件木马成功植入并触发后会对宿主 IC/IP 造成影响,尤其是在 IC/IP 运行阶段.但是,目前的研究中却很少有对故障现场的恢复机制进行考虑.可信执行等方式能够保障系统运行的可靠性,而恢复机制则有助于在木马激活时消除其对宿主 IC/IP 以及宏组件/实体的影响.

因此,需要我们根据硬件木马的影响特性制定相应的故障现场恢复机制,以提高 IC/IP 的安全性.

6.5 系统级漏洞分析

如图 21(a)所示,传统的 IC 设计流程首先考虑特定需求及功能实现,最后执行安全评估,存在严重的脆弱性[1-4].为此,Bernguier-Boher 等人[158]提出漏洞分析必须在 IC 设计的需求说明之后执行,并将分析结论加入到先前的规范中,实现安全策略在 IC 设计中的直接加入,从而对传统设计流程进行修改,如图 21(b)所示.



(a) 传统的IC设计流程[158]

(b) 执行漏洞分析的IC设计流程[158]

(c) 安全IC设计流程

图 21　改进的 IC 设计流程

此外，4.5 节中曾涉及到一种提高 IC 设计流程安全性的策略. 如图 21(c)所示，将传统的以功能实现为主、安全评估为辅的 IC 设计流程修改为以安全实现为主、兼顾功能实现的强安全 IC 设计流程(以 SoC 为例)，尽可能阻止攻击发生，提高攻击成功门槛，减少 IC 设计中的漏洞.

但是，执行全面漏洞分析与评估的成功率在很大程度上依赖于工程师与设计团队的经验、资金以及所采用的分析与评估策略等，主观性较强，缺乏一种系统的方法来对木马攻击进行有效分析与评估，因此需要探索全面的系统级漏洞的分析方法.

6.6　制定有效的度量标准

目前，对于硬件木马技术的评估与分析工作大部分都是针对旁路分析与增强检测技术展开的，而对于其它如硬件木马攻击、可信设计防范等技术的评估研究则相对较少. 已有的评估工作进展中所使用的性能评价指标种类多样、并不统一. 为比较不同的硬件木马攻击与安全防御策略，我们需要设计一套标准的评价指标. 这样的度量标准有助于形成综合的木马分类标准，更全面的分析木马防护技术. 因此，需要我们建立一套统一的度量标准，来对硬件木马各技术进行有效的评估与分析.

6.7　三维 IC 中的安全问题

随着先进半导体工艺及制造技术的发展，IC 设计公司正在逐步采用具有硅通孔(Through Silicon Vias，TSV)技术的 3DIC 设计技术来满足当前对 IC 高性能、低成本的要求[1,8,159-160]. 3DIC 封装可以在不同的工艺节点处容纳多个不同的模具，这可能会推迟用于所有相同功能的新进程节点这一重大决策的到来①.

同时，3DIC 的新开发流程需要一种新的 IC 供应链模式，这也为硬件木马攻击提供了新的机会[159]. 例如，第 3 节中曾提到 Hasan 等人[31]提出了一种利用 3DIC 独特结构的硬件木马. 该木马仅仅利用 3DIC 中间层的热效应来触发. 因此，需要对 3DIC 中的硬件木马进行更多的研究.

6.8　宏组件/实体中的硬件木马威胁

研究人员对 IC 级硬件木马技术研究已久，而对于比如 COTS 组件、印制电路板(Printed Circuit Board，PCB)[161-162]等此类宏观层面上硬件系统的硬件木马攻击却鲜有关注.

受研发成本与预算开销的影响，由第三方集成制造的 PCB 实体以及 COST 组件目前正被军事、金融、计算机、航空航天以及交通等②许多应用系统所广泛使用[163]，但其设计、实施及制造细节却难以追踪. 攻击者可能在这些宏组件/实体中植入硬件木马或是恶意修改其中的固件程序. 如果将它们部署到任务关键型应用型系统中，将会导致难以估计的严重后果[164-166].

因此，第三方组件/实体的安全性也值得我们关注. 有两种可能的方法来建立一个基于不可信组件或实体的安全系统. 第一类解决方案是对其进行认证，确保它们在部署或使用之前没有木马[1]. 而对于 PCB 和 COTS 等此类宏组件/实体，受其不可追踪特性及内部详细设计信息对维护者不可见性的限制，对它们的验证工作非常具有挑战性. 我们可以对其执行大量的各种功能和参数测试，或是通过对其执行结构与功能分析[69]，以验证是否满足既定的安全属性要求. 例如，Guo 等人[165,167-168]尝试利用 JTAG(Joint Test Action Group)接口创建板级环形振荡器网络(Ring Oscillator Network，RON)，以实现对 PCB 板进行强身份认证.

第二类解决方案是采用安全体系架构[145,148]，用以实现可能具有硬件木马的不可信 PCB 或 COTS 组件执行可信操作. 例如，4.5 节中提到的 SAFER PATH 结构，尝试利用可信执行中的投票机制应对 COTS 组件中的硬件木马攻击威胁[141].

除此之外，可以参考已经开发的利用可信执行方法来解决不可信 3PIP 核问题的思路，完成对

① Cadence. 3D ICs with TSVs—design challenges and requirements. http://www.europractice.stfc.ac.uk/vendors/cadence_3D_wp.pdf, 2011,01,07

② Chidley A. Use COTS parts to cut costs in military and aerospace systems. http://electronicdesign.com/components/use-cots-parts-cut-costs-military-andaerospace-systems, 2014, 05,04

PCB 或 COTS 组件的认证以及在硬件木马存在情况下的可信执行.

6.9 引入机器学习方法

随着机器学习技术研究的迅速深入,其应用方向也在不断扩张,一些研究人员尝试将机器学习方法引入到硬件木马检测研究中,进一步增强设计的安全性,提高木马检测的灵敏度.他们首先对硬件木马检测问题进行重新描述,将硬件木马检测问题转换为对 IC 设计的分类问题,然后对 IC 设计的参数特征等进行采样、特征提取、数据训练,最后执行分类等操作[77-80].例如,4.1.2 节中提到的 Li 等人利用有监督学习方法来进行硬件木马检测等.

机器学习方法的引入,能够带动硬件木马的研究朝着更加智能化、自动化的方向发展,但当前的一些进展中仍然存在着一些不足,这也激励着更多研究人员投入到这方面的研究中.

6.10 新兴的人工智能芯片

作为人工智能与芯片两大领域的交叉点,人工智能芯片正引来各方关注.各大芯片厂商也纷纷向人工智能芯片方向努力,积极研发并推出自主可控的人工智能芯片,并且已经在云端和终端等方面投入使用.较之于传统架构下的芯片,人工智能芯片具有高效的架构动态可变性、强大的数据处理与计算能力等优点.

同时,人工智能芯片的兴起需要新的生态系统与体系架构模型作为支撑,这也为硬件木马攻击提供了新的机会与场景.在面向人工智能片的生态系统与架构模型构建的过程中,一些潜在的漏洞可能会被攻击者利用,从而威胁人工智能芯片的安全.例如,一些原有的或新的硬件木马仍可以从体系架构层来威胁人工智能芯片的安全性,而研究人员往往忽略这方面的研究.

7 总 结

硬件木马成为当前信息系统领域的热点问题,本文对硬件木马的国内外研究成果和最新工作进行了整理归纳;讨论了硬件木马研究中的关键问题.针对硬件木马的研究热点内容进行了详细的对比总结,分析了当前研究工作的成果及存在的不足,讨论了硬件木马及相关技术的发展趋势,有助于清晰地展示目前研究工作所面临的挑战,促进硬件木马检测与防护技术的研究与发展,并为想从事硬件木马及相关技术研究的人员提供了指导.

参 考 文 献

[1] Xiao K, Forte D, Jin Y, et al. Hardware Trojans: Lessons learned after one decade of research. ACM Transactions on Design Automation of Electronic Systems, 2016, 22(1): 6:1-6:23

[2] Antonopoulos A, Kapatsori C, Makris Y. Trusted analog/mixed-signal/RF ICs: A survey and a perspective. IEEE Design & Test, 2017, 34(6): 63-76

[3] Bhunia S, Hsiao M S, Banga M, et al. Hardware Trojan attacks: Threat analysis and countermeasures. Proceedings of the IEEE, 2014, 102(8): 1229-1247

[4] Lv Y Q, Zhou Q, Cai Y C, et al. Trusted integrated circuits: The problem and challenges. Journal of Computer Science & Technology, 2014, 29(5): 918-928

[5] Adee S. The hunt for the kill switch. IEEE Spectrum, 2008, 45(5): 34-39

[6] Tehranipoor T, Koushanfar F. A survey of hardware Trojan taxonomy and detection. IEEE Design & Test, 2010, 27(1): 10-25

[7] Zhao Jian-Feng, Shi Gang. A survey on the studies of hardware Trojan. Journal of Cyber Security, 2017, 2(1): 74-90 (in Chinese)
(赵剑锋, 史岗. 硬件木马研究动态综述. 信息安全学报, 2017, 2(1): 74-90)

[8] Bhasin S, Regazzoni F. A survey on hardware Trojan detection techniques//Proceedings of the 2015 IEEE International Symposium on Circuits and Systems (ISCAS). Lisbon, Portugal, 2015: 2021-2024

[9] Karri R, Rajendran J, Rosenfeld K, et al. Trustworthy hardware: Identifying and classifying hardware Trojans. Computer, 2010, 43(10): 39-46

[10] Zhao Yi-Qiang, He Jia-Ji, Yang Song, et al. Research on defense technology hardware Trojans in integrated against circuits. Computer Engineering, 2016, 42(1): 128-132 (in Chinese)
(赵毅强, 何家骥, 杨松等. 集成电路中硬件木马防御技术研究. 计算机工程, 2016, 42(1): 128-132)

[11] Jacob N, Merli D, Heyszl J, et al. Hardware Trojans: Current challenges and approaches. IET Computers & Digital Techniques, 2014, 8(6): 264-273

[12] Tehranipoor M, Wang C. Introduction to Hardware Security and Trust. New York, USA: Springer Publishing Company, 2012: 195-229

[13] Sumathi G, Srivani L, Murthy D T, et al. A review on HT attacks in PLD and ASIC designs with potential defence solutions. IETE Technical Review, 2017, 34(1): 1-14

[14] Wang X, Tehranipoor M, Plusquellic J. Detecting malicious inclusions in secure hardware: Challenges and solutions//Proceedings of the 2008 IEEE International Workshop Hardware-Oriented Security Trust. Anaheim, USA, 2008: 15-19

[15] Moein S, Khan S, Gulliver T A, et al. An attribute based classification of hardware Trojans//Proceedings of the 2015 10th International Conference on Computer Engineering System. Piscataway, Egypt, 2015: 351-356

[16] Moein S, Gulliver T A, Gebali F, et al. A new characterization of hardware Trojans. IEEE Access, 2016, 4: 2721-2731

[17] Agrawal D, Baktir S, Karakoyunlu D, et al. Trojan detection using IC fingerprinting//Proceedings of the 2007 IEEE Symposium on Security and Piracy(SP'07). Berkeley, USA, 2007: 296-310

[18] Wang Xiao-Han, Li Xiong-Wei, Zhang Yang, et al. Hardware Trojan detection method based on kernel principal component analysis. Computer Measurement & Control, 2016, 24(1): 196-198(in Chinese)
(王晓晗，李雄伟，张阳等. 基于核主成分分析的硬件木马检测方法研究. 计算机测量与控制，2016，24(1)：196-198)

[19] Li Xiong-Wei, Wang Xiao-Han, Zhang Yang, et al. A new hardware Trojan detection method based on kernel maximum margin criterion. Acta Electronica Sinica, 2017, 45(3): 656-661(in Chinese)
(李雄伟，王晓晗，张阳等. 一种基于核最大间距准则的硬件木马检测新方法. 电子学报，2017，45(3)：656-661)

[20] Zhao Yi-Qiang, Liu Shen-Feng, He Jia-Ji, et al. Hardware Trojan detection technology based on self-organizing competition neural network. Journal of Huazhong University of Science and Technology (Natural Science Edition), 2016, 42(2): 51-55(in Chinese)
(赵毅强，刘沈丰，何家骥等. 基于自组织竞争神经网络的硬件木马检测方法. 华中科技大学学报(自然科学版)，2016，42(2)：51-55)

[21] Zhao Yi-Qiang, Yang Song, He Jia-Ji, et al. Hardware Trojan detection method based on principal component analysis. Journal of Huazhong University of Science and Technology (Natural Science Edition), 2015, 43(8): 66-69(in Chinese)
(赵毅强，杨松，何家骥等. 基于主成分分析的硬件木马检测方法. 华中科技大学学报(自然科学版)，2015，43(8)：66-69)

[22] Xue Ming-Fu, Hu Ai-Qun, Wang Jian. A novel hardware Trojan detection technique using heuristic partition and test pattern generation. Acta Electronica Sinica, 2016, 44(5): 1132-1138(in Chinese)
(薛明富，胡爱群，王箭. 基于探索式分区和测试向量生成的硬件木马检测方法. 电子学报，2016，44(5)：1132-1138)

[23] Xue Ming-Fu, Hu Ai-Qun, Liu Wei, et al. Detecting hardware Trojan through feature extraction in subspace domain. Journal of Southeast University (Natural Science Edition), 2014, 44(3): 457-461(in Chinese)
(薛明富，胡爱群，刘威等. 基于子空间域特征提取的硬件木马检测方法. 东南大学学报(自然科学版)，2014，44(3)：457-461)

[24] Fern N, Kulkarni S, Cheng K T T. Hardware Trojans hidden in RTL don't cares—Automated insertion and prevention methodologies//Proceedings of the 2015 IEEE International Test Conference(ITC). Anaheim, USA, 2015: 1-8

[25] Chakraborty R S, Paul S, Bhunia S. On-demand transparency for improving hardware Trojan detectability//Proceedings of the 2008 IEEE International Workshop on Hardware-Oriented Security and Trust (HOST'08). Anaheim, USA, 2008: 48-50

[26] Dupuis S, Ba P S, Flottes M L, et al. New testing procedure for finding insertion sites of stealthy hardware Trojans//Proceedings of the 2015 Design, Automation & Test in Europe Conference & Exhibition(DATE). Grenoble, France, 2015: 776-781

[27] Kamhoua C A, Zhao H, Rodriguez M, et al. A game-theoretic approach for testing for hardware Trojans. IEEE Transactions on Multi-Scale Computing Systems, 2017, 2(3): 199-210

[28] Saad W, Sanjab A, Wang Y P, et al. Hardware Trojan detection game: A prospect-theoretic approach. IEEE Transactions on Vehicular Technology, 2017, 66(9): 7697-7710

[29] Yang K, Hicks M, Dong M, et al. A2: Analog malicious hardware//Proceedings of the 2016 IEEE Symposium on Security and Piracy(SP'16). San Jose, USA, 2016: 18-37

[30] Patil V C, Vijayakumar A, Kundu S. Manufacturer turned attacker: Dangers of stealthy Trojans via threshold voltage manipulation//Proceedings of the 2017 IEEE North Atlantic Test Workshop(NATW). Providence, USA, 2017: 1-6

[31] Hasan S R, Mossa S F, Elkeelany O S A, et al. Tenacious hardware Trojans due to high temperature in middle tiers of 3-D ICs//Proceedings of the 2015 IEEE 58th International Midwest Symposium on Circuits and Systems (MWSCAS). Fort Collins, USA, 2015: 1-4

[32] Shiyanovskii Y, Wolff F, Rajendran A, et al. Process reliability based Trojans through NBTI and HCI effects//Proceedings of the 2010 NASA/ESA Conference on Adaptive Hardware and Systems (AHS). Anaheim, USA, 2010: 215-222

[33] Zhang X, Xiao K, Tehranipoor M J, et al. A study on the effectiveness of Trojan detection techniques using a red team blue team approach//Proceedings of the 2013 IEEE 31st VLSI Test Symposium(VTS'13). Berkeley, USA, 2013: 1-3

[34] Hu N H, Ye M M, Wei S. Surviving information leakage hardware Trojan attacks using hardware isolation. IEEE Transactions on Emerging Topics in Computing, DOI: 10.1109/TETC.2017.2648739

[35] Chang D, Bakkaloglu B, Ozev S. Enabling unauthorized RF transmission below noise floor with no detectable impact on primary communication performance//Proceedings of the 2015 IEEE 33rd VLSI Test Symposium (VTS'15). Napa, USA, 2015: 1-4

[36] Sepulveda J, Gross M, Zankl A, et al. Exploiting bus communication to improve cache attacks on systems-on-chips//Proceedings of the 2017 IEEE Computer Society Annual Symposium on VLSI (ISVLSI). Bochum, Germany, 2017: 284-289

[37] Fern N, San I, Cheng K T. Hardware Trojans in incompletely specified on-chip bus systems//Proceedings of the 2016 Design, Automation & Test in Europe Conference & Exhibition(DATE). Dresden, Germany, 2016: 527-530

[38] Tsoutsos N G, Maniatakos M. Fabrication attacks: Zero-overhead malicious modifications enabling modern microprocessor privilege escalation. IEEE Transactions on Emerging Topics in Computing, 2014, 2(1): 81-93

[39] Swierczynski P, Fyrbiak M, Koppe P, et al. FPGA Trojans through detecting and weakening of cryptographic primitives. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2015, 34(8): 1236-1249

[40] Li J, Sui Q, Chi M, et al. Design of a chip destructible hardware Trojan//Proceedings of the 2016 International Symposium on Computer, Consumer and Control (IS3C). Xi'an, China, 2016: 648-651

[41] Cha B, Gupta S K. A resizing method to minimize effects of hardware Trojans//Proceedings of the 2014 IEEE 23rd Asian Test Symposium(ATS'14). Hangzhou, China, 2014: 192-199

[42] Salmani H, Tehranipoor M, Karri R. On design vulnerability analysis and trust benchmarks development//Proceedings of the 2013 IEEE 31st International Conference on Computer Design(ICCD'13). Asheville, USA, 2013: 471-474

[43] Sudeendra K K, Sahoo S, Mahapatra A, et al. Analysis of side-channel attack AES hardware Trojan benchmarks against countermeasures//Proceedings of the 2017 IEEE Computer Society Annual Symposium on VLSI (ISVLSI'17). Bochum, Germany, 2017: 574-579

[44] Kumar K S, Chanamala R, Sahoo S R, et al. An improved AES hardware Trojan benchmark to validate Trojan detection schemes in an ASIC design flow//Proceedings of the 2015 19th International Symposium on VLSI Design and Test (VDAT'15). Ahmedabad, India, 2015: 1-6

[45] Zhang X H, Tehranipoor M. Case study: Detecting hardware Trojans in third-party digital IP cores//Proceedings of the 2011 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST'11). San Diego, USA, 2011: 67-70

[46] Sullivan D, Biggers J, Zhu G D, et al. Fight-metric: Functional identification of gate-level hardware trustworthiness//Proceedings of the 2014 51st ACM/EDAC/IEEE Design Automation Conference (DAC). San Francisco, USA, 2014: 1-4

[47] Hicks M, Finnicum M, King S T, et al. Overcoming an untrusted computing base: Detecting and removing malicious hardware automatically//Proceedings of the 2010 IEEE Symposium on Security and Privacy (SP'10). Berkeley/Oakland, USA, 2010: 159-172

[48] Zhou E R, Li S Q, Chen J H, et al. A novel detection method for hardware Trojan in third party IP cores//Proceedings of the 2016 International Conference on Information System and Artificial Intelligence (ISAI). Hong Kong, China, 2017: 528-532

[49] Salmani H, Tehranipoor M. Analyzing circuit vulnerability to hardware Trojan insertion at the behavioral level//Proceedings of the 2013 IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems(DFT'13). New York, USA, 2013: 190-195

[50] Waksman A, Suozzo M, Sethumadhavan S. FANCI: Identification of stealthy malicious logic using Boolean functional analysis//Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security(CCS'13). Berlin, Germany, 2013: 697-708

[51] Qiu Yi X, Li H W, Wang T C, et al. Property coverage analysis based trustworthiness verification for potential threats from EDA tools//Proceedings of the 2016 IEEE 25th Asian Test Symposium (ATS). Hiroshima, Japan, 2016: 43-48

[52] Oya M, Shi Y H, Yanagisawa M, et al. A score-based classification method for identifying hardware-Trojans at gate-level netlists//Proceedings of the 2015 Design, Automation & Test in Europe Conference & Exhibition (DATE'15). Grenoble, France, 2015: 465-470

[53] Haider S K, Jin C, Ahmad M, et al. Advancing the state-of-the-art in hardware Trojan detection. IEEE Transactions on Dependable and Secure Computing, 2019, 16(1): 18-32

[54] Rathmair M, Schupfer F, Krieg C. Applied formal methods for hardware Trojan detection//Proceedings of the 2014 IEEE International Symposium on Circuits and Systems (ISCAS'14). Melbourne VIC, Australia, 2014: 169-172

[55] Rajendran J, Vedula V, Karri R. Detecting malicious modifications of data in third-party intellectual property cores//Proceedings of the 52nd ACM/EDAC/IEEE Design Automation Conference(DAC'15). San Francisco, USA, 2015: 1-6

[56] Farahmandi F, Huang Y, Mishra P. Trojan localization using symbolic algebra//Proceedings of the 2017 22nd Asia and South Pacific Design Automation Conference(ASP-DAC). Chiba, Japan, 2017: 591-597

[57] Rathmair M, Schupfer F. Metrics for formal property checking against undesired circuit behavior in embedded systems//Proceedings of the ANALOG 2016; 15. ITG/GMM-Symposium ANALOG 2016. Bremen, Germany, 2016: 64-69

[58] Veeranna N, Schafer B. Hardware Trojan detection in behavioral intellectual properties (IP's) using property checking techniques. IEEE Transactions on Emerging Topics in Computing, 2017, 5(4): 576-585

[59] Portillo J, John E, Narasimhan S. Building trust in 3PIP using asset-based security property verification//Proceeding of the 2016 IEEE 34th VLSI Test Symposium (VTS'16). Las Vegas, USA, 2016: 1-6

[60] Rajendran J, Vedula V, Karri R. Formal security verification of third party intellectual property cores for information leakage//Proceedings of the 2016 29th International VLSI Design and 2016 15th International Conference on Embedded Systems(VLSID). Kolkata, India, 2016: 547-552

[61] Jin Y, Guo X, Dutta R G, et al. Data secrecy protection through information flow tracking in proof-carrying hardware IP—Part I: Framework fundamentals. IEEE Transactions on Information Forensics and Security, 2017, 12(10): 2416-2429

[62] Bidmeshki M M, Guo X, Dutta R G, et al. Data secrecy protection through information flow tracking in proof-carrying hardware IP—Part II: Framework automation. IEEE Transactions on Information Forensics and Security, 2017, 12(10): 2430-2443

[63] Chen X M, Liu Q Y, Yao S, et al. Hardware Trojan detection in third-party digital intellectual property cores by multi-level feature analysis. IEEE Transactions on Computer-Aided Design of Integrated Circuits Systems, 2018, 37(7): 1370-1383

[64] Courbon F, Loubet-Moundi P, Fournier J J A, et al. SEMBA: A SEM based acquisition technique for fast invasive hardware Trojan detection//Proceedings of the 2015 European Conference on Circuit Theory and Design (ECCTD). Trondheim, Norway, 2015: 1-4

[65] Bao C X, Forte D, Srivastava A. On application of one-class SVM to reverse engineering-based hardware Trojan detection //Proceedings of the 2014 15th International Symposium on Quality Electronic Design (ISQED). Santa Clara, USA, 2014: 47-54

[66] Bao C X, Forte D, Srivastava A. On reverse engineering-based hardware Trojan detection. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2015, 35(1): 49-57

[67] Nahiyan A, Xiao K, Yang K, et al. AVFSM: A framework for identifying and mitigating vulnerabilities in FSMs// Proceedings of the 53rd ACM/EDAC/IEEE Design Automation Conference (DAC'16). Austin, USA, 2016: 1-6

[68] Meade T, Zhang S J, Jin Y. Netlist reverse engineering for high-level functionality reconstruction//Proceedings of the 2016 21st Asia and South Pacific Design Automation Conference (ASP-DAC). Macau, China, 2016: 655-660

[69] Li W C, Wasson Z, Seshia S A. Reverse engineering circuits using behavioral pattern mining//Proceedings of the 2012 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST'12). San Francisco, USA, 2012: 83-88

[70] Voyiatzis A G, Stefanidis K G, Kitsos P. Efficient triggering of Trojan hardware logic//Proceedings of the 2016 IEEE 19th International Symposium on Design and Diagnostics of Electronic Circuits & Systems (DDECS). Kosice, Slovakia, 2016: 1-6

[71] Kitsos P, Simos D E, Torres-Jimenez J, et al. Exciting FPGA cryptographic Trojans using combinatorial testing// Proceedings of the 2015 IEEE 26th International Symposium on Software Reliability Engineering (ISSRE). Gaithersburg, USA, 2015: 69-76

[72] Xiao K, Zhang X H, Tehranipoor M. A clock sweeping technique for detecting hardware Trojans impacting circuits delay. IEEE Design & Test, 2013, 30(2): 26-34

[73] Aarestad J, Acharyya D, Rad R, et al. Detecting Trojans through leakage current analysis using multiple supply pad I_{DDQ} s. IEEE Transactions on Information Forensics and Security, 2010, 5(4): 893-904

[74] Nowroz A N, Hu K Q, Koushanfar F, et al. Novel techniques for high-sensitivity hardware Trojan detection using thermal and power maps. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2014, 33(12): 1792-1805

[75] Zhou B Y, Adato R, Zangeneh M, et al. Detecting hardware Trojans using backside optical imaging of embedded watermarks//Proceedings of the 2015 52nd ACM/EDAC/IEEE Design Automation Conference (DAC'15). San Francisco, USA, 2015: 1-6

[76] He J J, Zhao Y Q, Guo X L, et al. Hardware Trojan detection through chip-free electromagnetic side-channel statistical analysis. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2017, 25(10): 2939-2948

[77] Jap D, He W, Bhasin S. Supervised and unsupervised machine learning for side-channel based Trojan detection// Proceedings of the 2016 IEEE 27th International Conference Application-Specific Systems, Architectures and Processors (ASAP). London, UK, 2016: 17-24

[78] Liu Y, Jin Y, Nosratinia A, et al. Silicon demonstration of hardware Trojan design and detection in wireless cryptographic ICs. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2017, 25(4): 1506-1519

[79] Ni J, Li J, Lin S F, et al. A method of noise optimization for hardware Trojans detection based on BP neural network// Proceedings of the 2016 2nd IEEE International Conference on Computer and Communications (ICCC). Chengdu, China, 2017: 2800-2804

[80] Li J, Ni L, Chen J H, et al. A novel hardware Trojan detection based on BP neural network//Proceedings of the 2016 2nd IEEE International Conference on Computer and Communications (ICCC). Chengdu, China, 2017: 2790-2794

[81] Salmani H, Tehranipoor M, Plusquellic J. A novel technique for improving hardware Trojan detection and reducing Trojan activation time. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2012, 20(1): 112-125

[82] Zhou B, Zhang W, Thambipillai S, et al. Cost-efficient acceleration of hardware Trojan detection through fan-out cone analysis and weighted random pattern technique. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2016, 35(5): 792-805

[83] Hossain F S, Yoneda T, Inoue M. An effective scan segmentation approach to detect hardware Trojan in integrated circuits//Proceedings of the 2015 IEEE International WIE Conference on Electrical and Computer Engineering (WIECON-ECE). Dhaka, Bangladesh, 2016: 78-81

[84] Wei S, Potkonjak M. Malicious circuitry detection using fast timing characterization via test points//Proceedings of the 2013 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST'13). Austin, USA, 2013: 113-118

[85] Zarrinchian G, Zamani M S. Latch-based structure: A high resolution and self-reference technique for hardware Trojan detection. IEEE Transactions on Computers, 2016, 66(1): 100-113

[86] Guimaraes L A, Bastos R P, Fesquet L. Detection of layout-level Trojans by monitoring substrate with preexisting built-in sensors//Proceedings of the 2017 IEEE Computer Society Annual Symposium on VLSI (ISVLSI 2017). Bochum, Germany, 2017: 290-295

[87] Ismari D, Plusquellic J, Lamech C, et al. On detecting delay anomalies introduced by hardware Trojans//Proceedings of the 2016 35th IEEE/ACM International Conference on Computer-Aided Design (ICCAD). Austin, USA, 2017: 1-7

[88] Lodhi F K, Abbasi I, Khalid F, et al. A self-learning framework to detect the intruded integrated circuits//Proceedings of the 2016 IEEE International Symposium on Circuits and Systems (ISCAS). Montreal, Canada, 2016: 1702-1705

[89] Xue M F, Wang J, Hu A-Q. An enhanced classification-based golden chips-free hardware Trojan detection technique //Proceedings of the 2017 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST'17). Yilan, China, 2017: 1-6

[90] Liu Y, Huang K, Makris Y. Hardware Trojan detection through golden chip-free statistical side-channel fingerprinting //Proceedings of the 2014 51st ACM/EDAC/IEEE Design Automation Conference (DAC'14). San Francisco, USA, 2014: 1-6

[91] Yasin M, Rajendran J J, Sinanoglu O, et al. On improving the security of logic locking. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2016, 35(9): 1411-1424

[92] Kamakar R, Prassad N, Chattopadhyay S, et al. A new logic encryption strategy ensuring key interdependency//Proceedings of the 2017 30th International VLSI Design and 2017 16th International Conference on Embedded Systems (VLSID). Hyderabad, India, 2017: 429-434

[93] Rose G S. A chaos-based arithmetic logic unit and implications for obfuscation//Proceedings of the 2014 IEEE Computer Society Annual Symposium on VLSI (ISVLSI). Tampa, USA, 2014: 54-58

[94] Rajendran J, Zhang H, Zhang C, et al. Fault analysis-based logic encryption. IEEE Transactions on Computers, 2015, 64(2): 410-424

[95] Yasin M, Mazumdar B, Rajendran J J V, et al. SARLock: SAT attack resistant logic locking//Proceedings of the 2016 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST'16). McLean, USA, 2016: 236-241

[96] Samimi M S, Aerabi E, Kazemi Z, et al. Hardware enlightening: No where to hide your hardware Trojans!//Proceedings of the 2016 IEEE 2nd International Symposium on On-Line Testing and Robust System Design (IOLTS'16). Sant Feliu de Guixols, Spain, 2016: 251-256

[97] Subramanyan P, Ray S, Malik S. Evaluating the security of logic encryption algorithm//Proceedings of the 2015 International Symposium on Hardware-Oriented Security & Trust (HOST'15). Washington, USA, 2015: 137-143

[98] Juretus K, Savidis I. Reduced overhead gate level logic encryption//Proceedings of the 2016 26th International Great Lakes Symposium on VLSI. Boston, USA, 2016: 15-20

[99] Rajendran J, Pino Y, Sinanoglu O, et al. Logic encryption: A fault analysis perspective//Proceedings of the 2012 Design, Automation & Test in Europe Conference & Exhibition (DATE). Dresden, Germany, 2012: 953-958

[100] Wendt J B, Potkonjak M. Hardware obfuscation using PUF-based logic//Proceedings of the 2014 IEEE/ACM International Conference on Computer-Aided Design (ICCAD'14). San Jose, USA, 2014: 270-271

[101] Dunbar C, Qu G. Designing trusted embedded systems from finite state machines. ACM Transactions on Embedded Computing Systems, 2014, 13(5s): 1-20

[102] Liu B, Wang B. Reconfiguration-based VLSI design for security. IEEE Journal on Emerging and Selected Topics in Circuits and Systems, 2015, 5(1): 98-108

[103] Liu B, Wang B. Embedded reconfigurable logic for ASIC design obfuscation against supply chain attacks//Proceedings of the 2014 Conference on Design, Automation & Test in Europe. Dresden, Germany, 2014: 1-6

[104] Cocchi R P, Baukus J P, Chow L W, et al. Circuit camouflage integration for hardware IP protection//Proceedings of the 2014 51st ACM/EDAC/IEEE Design Automation Conference (DAC'14). San Francisco, USA, 2014: 1-5

[105] Rajendran J, Sam M, Sinanoglu O, et al. Security analysis of integrated circuit camouflaging//Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security (CCS'13). Berlin, Germany, 2013: 709-720

[106] Bi Y, Gaillardon P E, Hu X S, et al. Leveraging emerging technology for hardware security — Case study on silicon nanowire FETs and Graphene SymFETs//Proceedings of the 2014 IEEE 23rd Asian Test Symposium (ATS'14). Hangzhou, China, 2014: 342-347

[107] Massad M E, Garg S, Tripunitara V T. Integrated circuit (IC) decamouflaging: Reverse engineering camouflaged ICs within minutes//Proceedings of the 2015 Network and Distributed System Security Symposium. San Diego, USA, 2015: 1-14

[108] Xiao K, Forte D, Tehranipoor M. A novel built-in self-authentication technique to prevent inserting hardware Trojans. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2014, 33(12): 1778-1791

[109] Ba P S, Palanichamy M, Dupuis S, et al. Hardware Trojan prevention using layout-level design approach//Proceedings of the 2015 European Conference on Circuit Theory and Design (ECCTD). Trondheim, Norway, 2015: 1-4

[110] Ba P S, Dupuis S, Palanichamy M, et al. Hardware trust through layout filling: A hardware Trojan prevention technique//Proceedings of the 2016 IEEE Computer Society Annual Symposium on VLSI (ISVLSI). Pittsburgh, USA, 2016: 254-259

[111] Hossein-Talaee H, Jahanian A. Layout vulnerability reduction against Trojan insertion using security-aware white space

distribution//Proceedings of the 2017 IEEE Computer Society Annual Symposium on VLSI (ISVLSI'17). Bochum, Germany, 2017: 551-555

[112] Chakraborty R S, Pagliarini S, Mathew J, et al. A flexible online checking technique to enhance hardware Trojan horse detectability by reliability analysis. IEEE Transactions on Emerging Topics in Computing, 2017, 5(2): 260-270

[113] Narasimhan S, Yueh W, Wang X M, et al. Improving IC security against Trojan attacks through integration of security monitors. IEEE Design & Test of Computers, 2013, 29(5): 37-46

[114] Jin Y, Sullivan D. Real-time trust evaluation in integrated circuits//Proceedings of the 2014 Design, Automation and Test in Europe Conference and Exhibition (DATE'14). Dresden, Germany, 2014: 1-6

[115] Bao C X, Forte D, Srivastava A. Temperature tracking: Toward robust run-time detection of hardware Trojans. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2015, 34(10): 1577-1585

[116] Davoodi A, Li M, Tehranipoor M. A sensor-assisted self-authentication framework for hardware Trojan detection. IEEE Design & Test, 2013, 30(5): 74-82

[117] Guha K, Saha D, Chakrabarti A. Self aware SoC security to counteract delay inducing hardware Trojans at runtime//Proceedings of the 2017 30th International VLSI Design and 2017 16th International Conference on Embedded Systems (VLSID). Hyderabad, India, 2017: 417-422

[118] Jin Y, Maliuk D, Makris Y. Post-deployment trust evaluation in wireless cryptographic ICs//Proceedings of the 2012 Conference on Design, Automation and Test in Europe & Exhibition (DATE'12). Dresden, Germany, 2012: 965-970

[119] Liu Y, Volanis G, Huang K, et al. Concurrent hardware Trojan detection in wireless cryptographic ICs//Proceedings of the 2015 IEEE International Test Conference (ITC). Anaheim, USA, 2015: 1-8

[120] Liu C, Rajendran J, Yang C M, et al. Shielding heterogeneous MPSoCs from untrustworthy 3PIPs through security-driven task scheduling. IEEE Transactions on Emerging Topics in Computing, 2014, 2(4): 461-472

[121] Kalayappan R, Sarangi S R. SecCheck: A trustworthy system with untrusted components//Proceedings of the 2016 IEEE Computer Society Annual Symposium on VLSI (ISVLSI). Pittsburgh, USA, 2016: 379-384

[122] Gunti N B, Khatri A, Lingasubramanian K. Realizing a security aware triple modular redundancy scheme for robust integrated circuits//Proceedings of the 2014 22nd International Conference on Very Large Scale Integration (VLSI-SoC). Playa del Carmen, Mexico, 2014: 1-4

[123] Al-Anwar A, Alkabani Y, El-Kharashi M W, et al. Hardware Trojan detection methodology for FPGA//Proceedings of the 2013 IEEE Pacific Rim Conference on Communication, Computer and Signal Processing (PACRIM'13). Victoria, Canada, 2013: 177-182

[124] Reece T, Robinson W H. Detection of hardware Trojans in third party intellectual property using untrusted modules. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2016, 35(3): 357-366

[125] Rajendran J, Zhang H, Sinanoglu O, et al. High-level synthesis for security and trust//Proceedings of the 2013 IEEE 19th International On-Line Testing Symposium (IOLTS'13). Chania, Greece, 2013: 232-233

[126] Sengupta A, Bhadauria S, Mohanty S P. Low-cost security aware HLS methodology. IET Computers & Digital Techniques, 2017, 11(2): 68-79

[127] Rajendran J J, Sinanoglu O, Karri R. Building trustworthy systems using untrusted components: A high-level synthesis approach. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2016, 24(9): 2946-2959

[128] Sethumadhavan S, Waksman A, Suozzo M, et al. Trustworthy hardware from untrusted components. Communications of the ACM, 2015, 58(9): 60-71

[129] Fern N, San I, Koc C K, et al. Hiding hardware Trojan communication channels in partially specified SoC bus functionality. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2016, 36(9): 1435-1444

[130] Wehbe T, Wang X F. Secure and dependable NoC-connected systems on an FPGA chip. IEEE Transactions on Reliability, 2016, 65(4): 1852-1863

[131] Fery J, Yu Q Y. Exploiting state obfuscation to detect hardware Trojans in NoC network interfaces//Proceedings of the 2015 IEEE 58th International Midwest Symposium on Circuits and Systems (MWSCAS). Fort Collins, USA, 2015: 1-4

[132] Boraten T, Ditomaso D, Kodi A K. Secure model checkers for Network-on-Chip (NoC) architectures//Proceedings of the 2016 International Great Lakes Symposium on VLSI. Boston, USA, 2016: 45-50

[133] Boraten T, Kodi A K. Mitigation of denial of service attack with hardware Trojans in NoC architectures//Proceedings of the 2016 IEEE International Parallel and Distributed Processing Symposium. Chicago, USA, 2016: 1091-1100

[134] Kulkarni A, Pino Y, Mohsenin T. SVM-based real-time hardware Trojan detection for many-core platform//Proceedings of the 2016 17th International Symposium on Quality Electronic Design (ISQED). Santa Clara, USA, 2016: 362-367

[135] Kulkarni A, Pino Y, Mohsenin T. Adaptive real-time Trojan detection framework through machine learning//Proceedings of the 2016 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST'16). McLean, USA, 2016: 120-123

[136] Liu C L, Zhao Y Q, Shi Y F, et al. A system-on-chip bus architecture for hardware Trojan protection in security chips//Proceedings of the 2011 International Conference of Electron Devices and Solid-State Circuits (EDSSC). Tianjin, China, 2011: 1-2

[137] Kim L W, Villasenor J D. A system-on-chip bus architecture for thwarting integrated circuit Trojan horses. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2011, 19(10): 1921-1926

[138] Wahby R S, Howald M, Garg S, et al. Verifiable ASICs//Proceedings of the 2016 IEEE Symposium on Security and Piracy(SP'16). San Jose, USA, 2016: 1-18

[139] Shila D M, Venugopalan V, Patterson C D. Fides: Enhancing trust in reconfigurable based hardware systems//Proceedings of the 2015 IEEE High Performance Extreme Computing Conference (HPEC). Waltham, USA, 2015: 1-7

[140] Koushanfar F, Mirhoseini A. A unified framework for multimodal submodular integrated circuits Trojan detection. IEEE Transactions on Information Forensics and Security, 2011, 6(1): 162-174

[141] Beaumont M, Hopkins B, Newby T. Safer path: Security architecture using fragmented execution and replication for protection against Trojaned hardware//Proceedings of the 2012 Design, Automation & Test in Europe Conference & Exhibition (DATE'12). Dresden, Germany, 2012: 1000-1005

[142] Jin Y, Maliuk D, Makris Y. A post-deployment IC trust evaluation architecture//Proceedings of the 2013 IEEE 19th International On-Line Testing Symposium (IOLTS'13). Chania, Greece, 2013: 224-225

[143] Guo X L, Dutta R G, Jin X. Eliminating the hardware-software boundary: A proof-carrying approach for trust evaluation on computer systems. IEEE Transactions on Information Forensics and Security, 2017, 12(2): 405-417

[144] Wang X M, Zheng Y, Basak A, et al. IIPS: Infrastructure IP for secure SoC design. IEEE Transactions on Computers, 2015, 64(8): 2226-2238

[145] Kim L W, Villasenor J D. Dynamic function verification for system on chip security against hardware-based attacks. IEEE Transactions on Reliability, 2015, 64(4): 1229-1242

[146] Guha K, Saha D, Chakrabarti A. RTNA: Securing SoC architectures from confidentiality attacks at runtime using ART1 neural networks//Proceedings of the 2015 19th International Symposium on VLSI Design and Test (VDAT). Ahmedabad, India, 2015: 1-6

[147] Basak A, Bhunia S, Ray S, et al. Security assurance for system-on-chip designs with untrusted IPs. IEEE Transactions on Information Forensics and Security, 2017, 12(7): 1515-1528

[148] Liu C, Cronin P, Yang C M. A mutual auditing framework to protect IoT against hardware Trojans//Proceedings of the 2016 21st Asia and South Pacific Design Automation Conference(ASP-DAC). Macau, China, 2016: 69-74

[149] Johnson A, Chakraborty R S, Mukhopadhyay D. A PUF-enabled secure architecture for FPGA-based IoT applications. IEEE Transactions on Multi-Scale Computing Systems, 2015, 1(2): 110-122

[150] Hely D, Martin J, Triana G D P, et al. Experiences in side channel and testing based hardware Trojan detection//Proceedings of the 2013 IEEE 31st VLSI Test Symposium (VTS). Berkeley, USA, 2013: 1-4

[151] Reece T, Robinson W H. Analysis of data-leak hardware Trojan in AES cryptographic circuits//Proceedings of the 2013 IEEE International Conference on Technologies for Homeland Security(HST). Waltham, USA, 2013: 467-472

[152] Ni L, Li S, Chen J, et al. The influence on sensitivity of hardware Trojan detection by test vector//Proceedings of the 2014 Communications Security Conference (CSC). Beijing, China, 2014: 1-6

[153] Rithesh M, Harish G, Yellampalli S. Detection and analysis of hardware Trojan using dummy scan flip-flop//Proceedings of the 2015 International Conference on Smart Technologies and Management for Computing, Communication, Controls, Energy and Materials(ICSTM). Chennai, India, 2015: 439-442

[154] Kitsos P, Voyiatzis A G. FPGA Trojan detection using length-optimized ring oscillators//Proceedings of the 2014 17th Euromicro Conference on Digital System Design(DSD). Verona, Italy, 2014: 675-678

[155] Lecomte M, Foumier J J A, Maurine P. Thoroughly analyzing the use of ring oscillators for on-chip hardware Trojan detection//Proceedings of the 2015 17th International Conference on ReConFigurable Computing and FPGAs(ReConFig). Mexico City, Mexico, 2015: 1-6

[156] Kitsos P, Voyiatzis A G. A comparison of TERO and RO timing sensitivity for hardware Trojan detection applications//Proceedings of the 2015 Euromicro Conference on Digital System Design(DSD). Funchal, Portugal, 2015: 547-550

[157] Priya S R, Swetha P, Srigayathri D, et al. Hardware malicious circuit identification using self-referencing approach//Proceedings of the 2017 IEEE International Conference on Microelectronic Devices, Circuits and Systems(ICMDCS'17). Vellore, India, 2017: 1-5

[158] Bernguier-Boher N, Hely D, Beroulle V, et al. Increasing the security level of analog IPs by using a dedicated vulnerability analysis methodology//Proceedings of the 2013 14th International Symposium on Quality Electronic Design (ISQED). Santa Clara, USA, 2013: 531-537

[159] Xie Y, Bao C X, Serafy C, et al. Security and vulnerability implications of 3D ICs. IEEE Transactions on Multi-Scale Computing Systems, 2016, 2(2): 108-122

[160] Dofe J Y, Yu Q Y, Wang H L, et al. Hardware security threats and potential countermeasures in emerging 3D ICs//Proceedings of the 2016 International Great Lakes Symposium on VLSI. Boston, USA, 2016: 69-74

[161] Ghosh S, Basak A, Bhunia S. How secure are printed circuit boards against Trojan attacks? IEEE Design & Test, 2015, 32(2): 7-16

[162] Guo Z M, Di J, Forte D, et al. Obfuscation-based protection framework against printed circuit boards unauthorized operation and reverse engineering. ACM Transactions on Design Automation of Electronic Systems, 2017, 22(3): 54: 1-54:31

[163] Herr W. Keynote talk: Is it safe?//Proceedings of the 2015 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST'15). Washington, USA, 2015: 1-5

[164] Guo Z M, Tehranipoor M, Forte D, et al. Investigation of obfuscation-based anti-reverse engineering for printed circuit boards//Proceedings of the 2015 52nd ACM/EDAC/IEEE Design Automation Conference (DAC). San Francisco, USA, 2015: 1-6

[165] Guo Z M, Xu X L, Tehranipoor M T, et al. MPA: Model-assisted PCB attestation via board-level RO and temperature compensation//Proceedings of the 2017 IEEE Asian Hardware Oriented Security and Trust Symposium (AsianHOST). Beijing, China, 2017: 25-30

[166] Paley S, Hoque T, Bhunia S. Active protection against PCB physical tampering//Proceedings of the 2016 17th International Symposium on Quality Electronic Design (ISQED). Santa Clara, USA, 2016: 356-361

[167] Hennessy A, Zheng Y, Bhunia S. JTAG-based robust PCB authentication for protection against counterfeiting attacks//Proceedings of the 2016 21st Asia and South Pacific Design Automation Conference (ASP-DAC). Macau, China, 2016: 56-61

[168] Zhang F C, Hennessy A, Bhunia S. Robust counterfeit PCB detection exploiting intrinsic trace impedance variations//Proceedings of the 2015 IEEE 33rd VLSI Test Symposium (VTS'11). Napa, USA, 2011: 1-6

HUANG Zhao, born Ph. D. candidate. His main research interests include embedded system development, hardware security, hardware Trojan detection.

WANG Quan, Ph. D., professor, Ph. D. supervisor. His main research interests include embedded system development, hardware security, 3-D printing, and wireless networks.

YANG Peng-Fei, Ph. D., lecturer. His main research interests include embedded system architecture and security.

Background

Information Security is a hot topic which has been discussed in the electronic and computer domains for a long time. Competition between software designers and hackers has been intensifying since the 1980s, while the underlying hardware is generally considered secure. However, with the increasing development of mobile internet and intelligent terminals, smart devices have entered all aspects of our daily life. Security issues are no longer restricted to the software and cyber perspectives, electronic devices themselves may also be unsafe. In particular, the highly distributed nature of modern electronic devices in design, manufacturing, and distribution has incurred new attack sources, which provides opportunities for an adversary to maliciously tamper the supply chain. Such malicious modifications to the original designs are called hardware Trojan or backdoor. It can make the chip fault, reveal confidential information, performance degrading, or even permanently failing. Research on hardware Trojan has grown dramatically over the past decade and is expected to continue. Unfortunately, there exist some limitations in current investigation work. Therefore, we write this survey article to comprehensively describe the latest academic and industrial research work on hardware Trojan.

This article first discusses the relevant research work and state-of-the-art of hardware Trojan from three different aspects. Based on analyzing the main progress at home and abroad in detail, it induces several key problems such as hardware Trojan design, Trojan detection techniques, design for trust, runtime defenses, component-level protection, architecture-level protection, and so forth. Finally, it also discusses the future development trend of hardware Trojans.

This article belongs to the review research work in hardware Trojan domain. The purpose of this article is to show the current research progress, existing problems, and the possible future directions. It can also provide a general bird's eye view picture and a possible guide for those who want to engage in hardware Trojan research at the same time.

This work was supported by the National Natural Science Foundation of China under Grant Nos. 61572385, 61702395, 61711530248, the Project of Shaanxi Provincial Science and Technology Co-ordinator Innovative Engineering under Grant No. 2015KTCXSF-01, and completed under the guidance of Professor Quan Wang. Opinions, findings, conclusions and recommendations expressed in this material are those of the authors and may not reflect the views of the funding entities. Wang Quan is the corresponding author.