

# 基于节点活性的硬件木马检测方法

冯秋丽<sup>1,2</sup>, 侯 波<sup>1</sup>, 刘燕江<sup>2</sup>, 恩云飞<sup>1</sup>, 王力伟<sup>1</sup>

(1 工业和信息化部电子第五研究所 电子元器件可靠性物理及其应用技术重点实验室, 广东 广州 510610;

2 广东工业大学 材料与能源学院, 广东 广州 510006)

**摘 要:** 本文提出了一种基于节点活性的硬件木马检测方法, 针对电路中的低活性节点生成测试向量, 结合多参数旁路检测方法, 实现对硬件木马的检测。以 AES 为目标电路并植入硬件木马, 进行仿真及 FPGA 实验, 实验结果表明与随机测试向量相比, 本文生成的测试向量可将木马节点的翻转概率提高一个数量级、木马检测灵敏度分别提高 6.75%(仿真)、77.4%(FPGA), 硬件木马的检测精度达到  $10^{-4}$ 。

**关键词:** 芯片安全; 硬件木马检测; 节点活性; 测试向量生成

**中图分类号:** TN4

**文献标识码:** A

**文章编号:** 1000—7180(2017)01—0035—05

## A Novel Hardware Trojan Detection Method Based on Node Activity State

FENG Qiu-li<sup>1,2</sup>, HOU Bo<sup>1</sup>, LIU Yan-jiang<sup>1</sup>, EN Yun-fei<sup>1</sup>, WANG Li-wei<sup>1</sup>

(1 Science and Technology on Reliability Physics and Application of Electronic Component Laboratory,

The Fifth Electronic Research Institute of the Ministry of Industry and

Information Technology, Guangzhou 510610, China; 2 School of Material and Energy Resources,

Guangdong University of Technology, Guangzhou 510000, China)

**Abstract:** In this paper, we propose a hardware Trojan detection method based on nodes activity state, in view of the low activity nodes of original circuit generate test vectors. Combined with the method of multi-parameter side channel signals to detect Hardware Trojan. The proposed method is verified by simulation and FPGA experiment carried on AES original circuit which is implanted with hardware trojan. The results show that compared with random test vector, the proposed method can improve the transition probability of the trojans of nodes an order of magnitude and increase the sensitivity of the Hardware Trojan detection by 6.75%(simulation), 77.4%(FPGA), detect the Hardware Trojan whose equivalent area is as small as  $10^{-4}$  of the total size of the circuit.

**Key words:** IC security; node activity state; hardware trojan detection; test pattern generation

### 1 引言

随着半导体制造工艺的飞速发展, 电路规模的不断增大, 集成电路广泛应用在国民生活的各个领域。为了缩短集成电路的设计周期及降低成本, 芯片设计、制造、测试等阶段逐渐分离, 集成电路产业链变得并非完全可信, 如图 1 所示为集成电路产业链的可信任性分析。攻击者可以在图 1 所示的不可信任的环节中植入称之为硬件木马的恶意电路。硬件

木马可以在特定的状态下激活电路, 使电路系统出现暂时性瘫痪、芯片失效、功能紊乱或者信息泄露等状况。这些包含硬件木马的芯片一旦被应用在国家安全、军事、通信、金融、交通等安全敏感领域中, 势必会造成巨大的经济损失和意想不到的灾难<sup>[1]</sup>。

硬件木马由触发部分和有效载荷两部分组成, 基本结构如图 2 所示。硬件木马的触发部分决定植入的硬件木马何时开始工作, 而有效载荷决定木马被激活后实现何种功能。

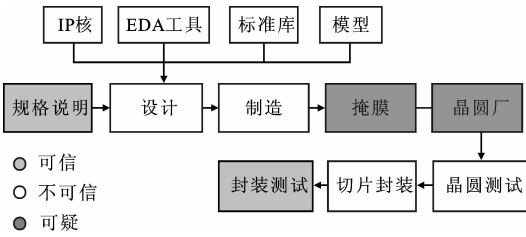


图 1 集成电路产业链的可信任分析

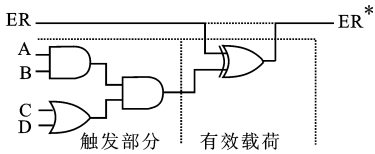


图 2 硬件木马的结构图

由于硬件木马的危害极大,硬件木马的检测方法成为国内外研究的重点.当前的硬件木马检测方法主要有三种:失效分析法、逻辑测试法和旁路分析法<sup>[2]</sup>.失效分析法是用精密的仪器设备扫描重构目标电路设计,通过比较反向工程设计与原始设计判断是否存在硬件木马.逻辑测试法是通过对比目标电路的响应与待测电路响应的差异来实现硬件木马的检测.旁路分析方法通过对比待测芯片与原始芯片的旁路信息来检测待测芯片中是否存在硬件木马<sup>[3]</sup>.是目前主流的硬件木马检测方法.然而旁路分析方法易受工艺波动的影响<sup>[4]</sup>,并且其检测灵敏度依赖于所施加的测试向量是否能最大程度地激活硬件木马.

为了有效地隐藏硬件木马,恶意攻击者一般选择电路中低活性节点作为木马植入点,低观测节点作为载荷部分<sup>[5]</sup>.基于此,本文提出了一种基于节点活性的硬件木马检测方法,针对基准电路中的低活性节点生成测试向量,结合旁路信号分析方法,利用多个参数之间的固有关系实现硬件木马的检测<sup>[6]</sup>.

2 基于活性度的硬件木马检测

2.1 检测原理

集成电路节点的活性可定义为节点的翻转概率,即节点值由 1 变换到 0,或由 0 变换到 1 的概率,节点  $i$  的翻转概率  $\theta_i$  定义如式(1):

$$\theta_i = \frac{m_i}{n} \tag{1}$$

式中,  $m_i$  是在施加测试向量后的一段时间内节点  $i$  的翻转次数,  $n$  是整个电路的节点翻转次数.集成电路动态电流与电路节点的翻转次数成正比.翻转次数越多,则其产生的动态电流越大.因此,对于节点

$i$ ,其翻转概率  $\theta_i$  越大,则该节点产生的动态电流也越大.

恶意攻击者为了使得植入的硬件木马难以检测,往往选择翻转概率低的节点植入硬件木马.因此这些低翻转概率节点即为可能的木马节点.低翻转概率节点由于很少翻转,因此旁路分析及逻辑测试难以检测到硬件木马.

由于恶意攻击者会将硬件木马植入到低翻转概率节点,因此硬件木马检测需要提高低翻转概率节点的翻转概率.假设低翻转概率节点  $j$  的翻转概率为  $\theta_j$ ,由于节点的翻转概率与所施加的测试向量有关,因此针对硬件木马生成的测试向量需要提高  $\theta_j$ .

假设原始向量组为  $(V_1, V_2, \dots, V_n)$ ,其对应的节点  $j$  的翻转概率为  $\theta_{j, V_m}$ ,则针对硬件木马测试的向量  $V_{HTH}$  应为最大翻转概率  $\theta_{j, V_m}$  所对应的向量  $V_m$ .

2.2 检测方法

根据 2.1 节的检测原理,本文提出的基于节点活性的硬件木马检测方法如图 3 所示.

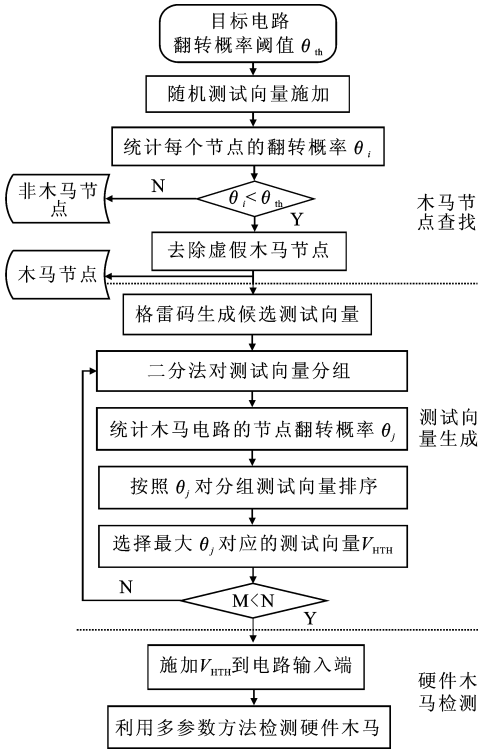


图 3 基于节点活性的硬件木马检测方法

图 3 所示的基于节点活性的硬件木马检测方法包含三部分:木马节点查找、测试向量生成及硬件木马检测.

(1) 木马节点查找

首先设置翻转概率阈值  $\theta_{th}$ ,其次对目标电路施

加大量的随机测试向量,统计电路中每个节点的翻转概率,筛选出小于翻转概率阈值的电路节点,从中去除虚假木马节点,则剩余的节点为硬件木马节点,其中虚假节点为节点状态值始终为“0”或“1”的节点。

(2) 测试向量生成

采用格雷码生成候选测试向量,格雷码的所有相邻整数之间只有一个数字不同,减少了由一个状态到下一个状态是的逻辑混淆,本文利用这种编码方式生成候选测试向量,有效避免由输入状态切换引起的不必要的电流脉冲,增加了由木马模块活动产生的额外电流增益。其次采用二分法对候选测试向量分组,将每组测试向量的个数记为  $M$ ,统计每组测试向量下木马节点的翻转概率,以翻转概率的大小对测试向量进行排序,挑选对应翻转概率最大的测试向量作为本次划分的测试向量。将最终划分的每组测试向量的个数记为  $N$ ,判断  $M$  是否大于  $N$ ,若  $M$  大于  $N$ ,则对测试向量继续进行划分,若  $M$  小于  $N$ ,则该向量被选取为硬件木马测试向量。 $N$  取值较小时,可能无法完全激活硬件木马, $N$  取值较大,木马检测的时间较长,综合考虑后,本文选取  $N = 5$ ,即测试向量最终分为 5 个一组。

(3) 硬件木马检测

将生成的测试向量施加到待测电路的输入端,结合旁路分析方法,利用动态电流  $I_{DDT}$  和电路的特征频率  $F_{MAX}$  来去除工艺偏差的影响,实现对硬件木马的检测。

3 实验验证

3.1 仿真步骤

本文以密钥长度为 128 bit 高级加密(AES)电路作为目标电路。采用 Synopsys 公司的 Design Compiler 综合工具在 SMIC180nm 工艺下对 RTL 级电路进行综合,将综合后的电路转化为 spice 格式的电路网表,采用 Nanosim 仿真工具来仿真电路的瞬态电流和最大工作频率。并采用蒙特卡罗方法来模拟不同芯片间的工艺噪声。具体仿真过程如图 4 所示。

以 AES 电路为基准电路,在其中植入如图 5 所示的计数器型硬件木马。当计数达到预设值,如  $2^k - 1$  时,硬件木马被触发,电路的输出值发生改变,导致电路功能紊乱。

利用 2 位、16 位计数器作为硬件木马电路的触发部分,选择电路的低活性节点作为硬件木马的触

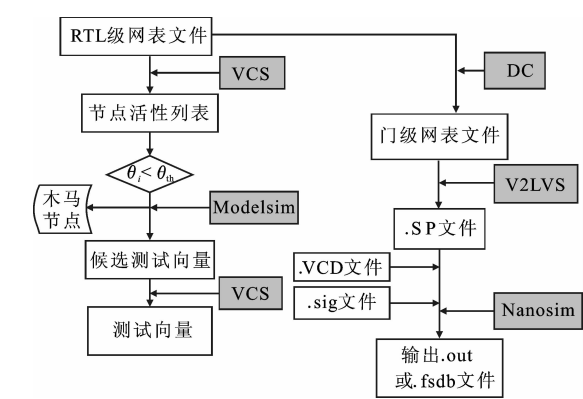


图 4 硬件木马检测的仿真流程图

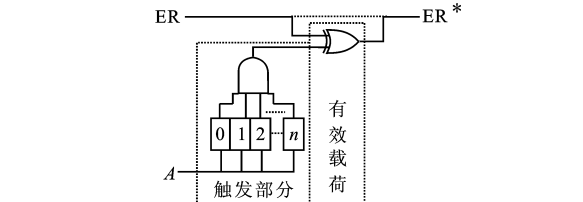


图 5 计数器型硬件木马的结构

发端,有效载荷部分由一个异或门构成。通过 Design Compiler 工具综合,得到两种不同面积的木马电路,植入有 2 位计数器的电路是  $CUT_1$ 、含有 16 位计数器的电路是  $CUT_2$ 。表 1 统计了木马模块与  $CUT_1$ 、 $CUT_2$  电路的面积比  $\gamma$ 。

表 1 硬件木马与  $CUT_1$ 、 $CUT_2$  电路的面积比

| 电路      | $\gamma$             |
|---------|----------------------|
| $CUT_1$ | $2.1 \times 10^{-4}$ |
| $CUT_2$ | $1.0 \times 10^{-3}$ |

3.2 实验结果

实验中选取随机测试向量施加到目标电路上,获取电路节点的翻转概率,图 6 为电路内部节点的翻转概率情况。由概率分布直方图可知:电路的节点翻转概率大多分布在区间  $1.2 \times 10^{-3} \sim 1.7 \times 10^{-3}$ ;电路节点活性分布不均匀,存在低“0”和低“1”活性节点;随着电路中节点数目增多,低活性节点的数目也在增加。

临界阈值的选取与低活性节点个数的关系如图 7 所示。临界阈值  $\theta_{th}$  的取值越小,满足低活性条件的节点越少,反之满足低活性条件的节点越多。临界阈值  $\theta_{th}$  决定硬件木马检测效率, $\theta_{th}$  取值较大时,木马检测的时间成本大, $\theta_{th}$  取值较小时,无法全面检测出硬件木马。

本文将临界阈值  $\theta_{th}$  取为  $0.35 \times 10^{-3}$ ,翻转概率小于临界阈值的节点列为可疑节点,去除重复及节

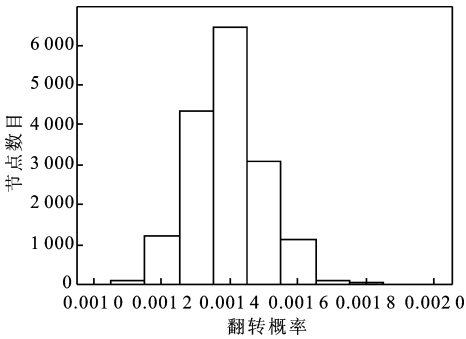


图 6 电路内部节点翻转情况统计

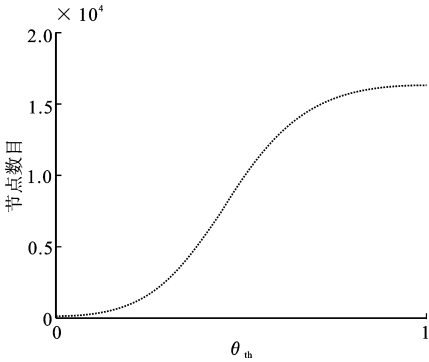


图 7 临界阈值与低活性节点个数的关系

点值固定不变的节点后,最终得出低活性节点的个数为 5 个,即木马节点有 5 个,如表 2 所示为对应的木马节点及节点翻转概率。

表 2 木马节点及节点的翻转概率

| 木马节点           | 翻转概率                  |
|----------------|-----------------------|
| u0/sa11[0]     | $0.35 \times 10^{-3}$ |
| u0/sa22[2]     | $0.34 \times 10^{-3}$ |
| u0/sa11_sr[5]  | $0.33 \times 10^{-3}$ |
| u0/sa10_mc[1]  | $0.34 \times 10^{-3}$ |
| u0/sa12_sub[5] | $0.33 \times 10^{-3}$ |

选取使得翻转率概率  $\theta_j$  取得最大值时对应的测试向量  $V_m$  并与随机测试向量  $V_r$  作对比,两种测试向量下木马节点的翻转概率情况如表 3 所示。

表 3 两种测试向量下三种电路的翻转情况

| 翻转概率  | AES                    | CUT <sub>1</sub>      | CUT <sub>2</sub>      |
|-------|------------------------|-----------------------|-----------------------|
| $V_m$ | $1.217 \times 10^{-4}$ | $1.06 \times 10^{-4}$ | $1.08 \times 10^{-4}$ |
| $V_r$ | $0.524 \times 10^{-5}$ | $0.51 \times 10^{-5}$ | $0.52 \times 10^{-5}$ |

由表 3 知本文生成的测试向量相较于随机测试向量,节点的翻转概率提高了一个数量级。

将生成的测试向量  $V_m$ 、 $V_r$  依次施加到待测电路上,采集动态电流  $I_{DDT}$  和特征频率  $F_{MAX}$ ,利用 3.1 节所示的仿真方法,结果如图 8 和图 9 所示。

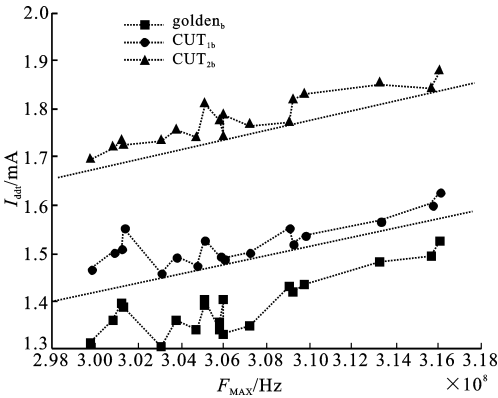


图 8 测试向量  $V_m$  激励下  $I_{DDT}$  与  $F_{MAX}$  的关系图

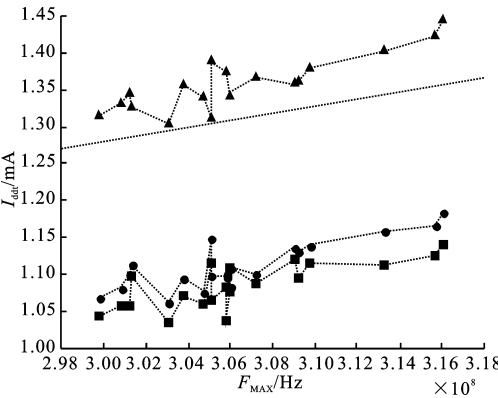


图 9 测试向量  $V_r$  激励下  $I_{DDT}$  与  $F_{MAX}$  的关系图

观察图 8 知,本文针对硬件木马生成的测试向量  $V_m$  可以完全检测出植入了硬件木马的芯片。将木马检测灵敏度  $S$  定义为待测电路的  $I_{DDT}$  与目标电路的  $I_{DDT}$  的差值与目标电路  $I_{DDT}$  的比值<sup>[7]</sup>,表达如式(2)所示。

$$S = \frac{I_{DDT,CUT} - I_{DDT,golden}}{I_{DDT,golden}} \quad (2)$$

观察图 9 可知,对电路施加随机测试向量进行硬件木马的检测无法分辨 CUT<sub>1</sub> 中是否被植入了硬件木马。两种测试向量激励下,硬件木马的检测灵敏度如表 4 所示。

表 4 两种测试向量激励下的灵敏度对比

| $S$              | $V_m/\%$ | $V_r/\%$ |
|------------------|----------|----------|
| CUT <sub>1</sub> | 9.28     | 2.53     |
| CUT <sub>2</sub> | 27.84    | 24.23    |

由表 4 知对于 CUT<sub>1</sub>,  $V_m$  对硬件木马检测的灵敏度比随机测试向量高出 6.75%,对于 CUT<sub>2</sub>,本文所选测试向量比随机向量的灵敏度高出 3.61%。纵向比较两种待测电路发现,在所选测试向量的激励下,待测电路 2 的检测灵敏度比待测电路 1 高

18.56%。由此可见,利用本文所述方法生成的测试向量不仅可以提高硬件木马节点翻转概率,也提高了旁路信号检测的灵敏度。

采用 FPGA 实验对本文提出的方法进行验证,使用 24 块同型号的 FPGA 芯片来模拟工艺波动,实验结果如图 10 和图 11 所示。

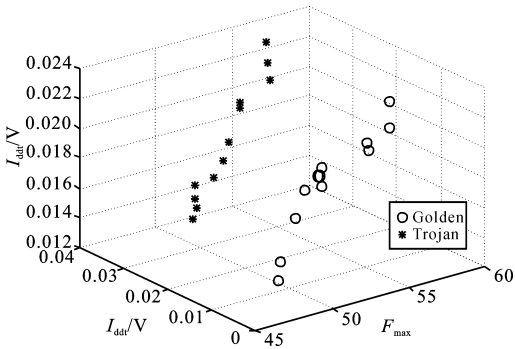


图 10 测试向量  $V_m$  激励下芯片检测结果

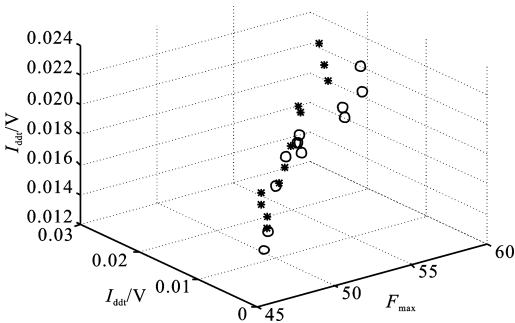


图 11 测试向量  $V_r$  激励下芯片检测结果

图 10 和图 11 分别是  $V_m$ 、 $V_r$  激励下多参数检测硬件木马的三维空间图。图 10 可以明显看出,采用本文所提出的硬件木马检测方法可以完全检测出硬件木马,且检测灵敏度相对于随机生成方法高出 77.47%。

4 结束语

本文以电路中的低活性节点作为硬件木马的触发节点,针对这些低活性节点生成测试向量,结合多参数旁路分析方法检测硬件木马。实验结果表明与随机测试向量相比,本文提出的方法可将硬件木

马的激活概率提高一个数量级、硬件木马检测灵敏度提高 77.4%,硬件木马的检测精度可达到  $10^{-4}$ 。接下来的研究中可以选择更高效的数据后处理算法进一步提高硬件木马的检测精度。

参考文献:

[1] 刘华锋, 罗宏伟, 王力纬. 硬件木马检测技术的研究 [C]// 2011 第十四届全国可靠性物理学术讨论会. 中国, 苏州, 2011.

[2] Wang X, Tehranipoor M, Plusquellic J. Detecting malicious inclusions in secure hardware: Challenges and solutions [C] // Hardware-Oriented Security and Trust, 2008. HOST 2008. IEEE International Workshop on. USA, Anaheim, IEEE, 2008; 15-19.

[3] Jin, Yier, Yiorgos Makris. Hardware Trojan detection using path delay fingerprint [C]// Hardware-Oriented Security and Trust, 2008. HOST 2008. IEEE International Workshop on. USA, Anaheim, IEEE, 2008.

[4] 赵崇征, 邓高明, 赵强. 基于旁路分析的集成电路芯片硬件木马检测 [J]. 微电子学与计算机, 2011, 28 (11): 5-9.

[5] Jha S, Jha S K, Randomization based probabilistic approach to detect trojan circuits [C]// High Assurance Symposium, 2008. China, Nanjing, IEEE, 2008; 117-124.

[6] Du D, Narasimhan S, Chakraborty R S, et al. Self-referencing: a scalable side-channel approach for hardware Trojan detection [C]// Proceedings of the 12th international conference on Cryptographic hardware and embedded systems. Santa Barbara, Springer-Verlage, 2010; 173-187.

[7] 尚子靖. 基于活性测度的硬件木马检测方法研究 [D]. 北京: 北京大学, 2012; 23-24.

作者简介:

冯秋丽 女, (1990-), 硕士研究生. 研究方向为硬件木马的检测.

侯 波 (通信作者) 男, (1985-), 工程师. 研究方向为硬件木马检测. E-mail: houbo@ceprei.com.