

# 基于 IDDQ 测试的 VLSI 门内电阻式桥接故障仿真

许爱强<sup>1</sup>, 唐小峰<sup>1,2</sup>, 牛双诚<sup>1</sup>, 杨智勇<sup>1,3</sup>

(1. 海军航空工程学院科研部, 烟台 264001; 2. 中国人民解放军 92514 部队, 烟台 264007;

3. 中国人民解放军 91370 部队, 福州 350015)

**摘要:** 为真实模拟集成电路中的桥接故障并评价测试集质量, 提出一种基于静态电源电流 (IDDQ) 测试的逻辑电路门内电阻式桥接故障仿真算法. 首先, 针对该故障类型, 提出一种覆盖率评价标准; 其次, 利用电路级故障注入与仿真方法构造基本逻辑门单元的故障字典; 最后, 通过在逻辑电路功能仿真中查询故障信息实现门级的故障仿真. 仿真实验表明: 相比于传统方法, 所提方法能更好地反映测试集对真实桥接故障的覆盖效果, 并具备良好的仿真效能.

**关键词:** 超大规模集成电路 (VLSI) 测试; 电阻式桥接故障; 静态电源电流 (IDDQ) 测试; 故障仿真; 故障覆盖率  
**中图分类号:** TP 391      **文献标志码:** A      **文章编号:** 0254-0037(2016)01-0128-06

**doi:** 10.11936/bjutxb2015040078

## Simulation of Intra-gate Resistive Bridging Faults in VLSI Based on IDDQ Testing

XU Aiqiang<sup>1</sup>, TANG Xiaofeng<sup>1,2</sup>, NIU Shuangcheng<sup>1</sup>, YANG Zhiyong<sup>1,3</sup>

(1. Department of Scientific Research, Naval Aeronautical and Astronautical University, Yantai 264001, China;

2. Unit 92514 of the PLA, Yantai 264007, China; 3. Unit 91370 of the PLA, Fuzhou 350015, China)

**Abstract:** To realistically simulate the bridging faults in VLSI and to evaluate the quality of the test set, an intra-gate non-zero resistance bridging fault simulation algorithm based on IDDQ testing was proposed. First, a fault coverage criteria was proposed for this type of fault. Second, the fault dictionary for every type of the primitive logic gate cells was constructed by using the circuit-level fault injection and simulation method. Lastly, the gate-level fault simulation was accomplished by querying the fault dictionary when performing the functional simulation of the target logic circuit. Experimental results show that, comparing with the traditional approach, the proposed method in this paper can better reflect the fault covering ability of the test set against the realistic bridging faults and the simulation is efficient.

**Key words:** VLSI testing; resistive bridging fault; IDDQ testing; fault simulation; fault coverage

应用真实的故障模型来评价超大规模集成电路 (very large scale integration, VLSI) 测试集质量一直以来都是 VLSI 测试领域的重要研究目标. 桥接故障 (bridging fault, BF) 是 VLSI 中最常见的故障之一, 尤其是随着电路集成密度的不断增加, 其重要性越发凸显. 许多相关研究假定桥接阻值为  $0 \Omega$ <sup>[1]</sup> 或

者为某一固定值<sup>[2]</sup>. 然而, 理论上实际桥接阻值可能分布于 0 到无穷大的动态区间上, 是一个随机变量. 这给 VLSI 的测试带来了新的挑战. 针对这一问题, 电阻式桥接故障 (resistive bridging fault, RBF) 在近年来得到广泛研究<sup>[3-5]</sup>. 按 RBF 发生位置的不同, 可以将其分为门间 (inter-gate) RBF 和门内

收稿日期: 2015-04-25

基金项目: “泰山学者”建设工程专项经费资助项目; 中国博士后科学基金资助项目 (2013M542535)

作者简介: 许爱强 (1963—), 男, 教授, 主要从事电子信息系统测试与诊断技术方面的研究, E-mail: vivorimage@126.com

(intra-gate) RBF 两类. 许多研究关注门间 RBF, 鲜有对门内 RBF 的研究报道. 然而, 考虑到: 1) 门内 RBF 是一种局部故障模型, 桥接故障点之间距离相近, 在实际电路中发生的可能性较高; 2) 由于被测电路 (circuit under test, CUT) 的最终布局实现往往难以获得, 门间 RBF 通常需要考虑所有的连线组合, 其计算复杂度为  $O(m^2)$ , 其中  $m$  为 CUT 中连线的数量<sup>[6]</sup>, 而门内 RBF 的计算复杂度要小得多, 为  $O(n)$ , 其中  $n$  为 CUT 中逻辑门的数量; 3) 门内 RBF 可达逻辑门电路中的内部节点 (internal nodes)<sup>[6]</sup>, 而门间 RBF 则不能. 因此, 研究 VLSI 的门内 RBF 具有现实意义. 目前, 研究 VLSI 桥接故障的方法主要有电压测试法 (逻辑测试) 和供电电流测试法 2 类. 前者需将故障的影响传播到 CUT 的原始输出端; 而后者需在 CUT 中设置额外的电流监控器. 有研究表明, 静态电源电流 (IDDQ) 测试对桥接故障的覆盖效果通常要好于逻辑测试<sup>[7]</sup>. 因此, 本文基于  $I_{DDQ}$  测试研究 VLSI 中的门内 RBF 仿真问题.

## 1 门内 RBF 及其覆盖率模型

这里研究门内 RBF (以下简称 RBF) 的目的是考察这类故障如何在单个逻辑门的局部范围内对整个被测电路 (circuit under test, CUT) 的  $I_{DDQ}$  特征产生影响. 在两点间桥接假设下, 一个 RBF 可被定义为  $f = \langle n_1, n_2, R \rangle$ , 其中  $n_1$  和  $n_2$  是被短接的电路节点, 它们既可以是外部连线也可以是内部节点;  $R$  为桥接阻值随机变量. 特别地, 当  $R = 0$  时, 即为理想桥接故障; 当  $R = C$  ( $C$  为固定常数) 时, 则为固定阻值桥接故障.

以图 1(a) 所示的两输入 CMOS 与非门为例, 一个 RBF:  $f = \langle L_A, L_C, R_B \rangle$  发生在连线  $L_A$  和内部节点  $N_C$  之间, 桥接阻值  $R = R_B$ . 假设给定测试激励  $t = "11"$  ( $L_A = '1', L_B = '1'$ ), 正常状态下 PMOS 管  $M_1$  和  $M_2$  截止, NMOS 管  $M_3$  和  $M_4$  导通. 由于晶体管截止电阻通常很大, 以至于  $I_{DDQ}$  值非常小 ( $\leq nA$  数量级). 然而, 由于桥接故障  $f$  的影响, 输入线  $L_A$  上的电平被拉低, 这可能致使以其作为输入控制端的晶体管  $M_2$  和  $M_3$  工作于非稳定的可变电阻区. 图 1(b) 显示了该与非门电路在  $f$  存在时的一个等效电路, 其中  $R_{UP}$  是前级驱动门的上拉网络等效电阻;  $R_{ON}$  是晶体管的导通电阻; 而晶体管的截止电阻则被忽略. 这样, 电路中就形成了导通  $V_{DD}$  和地的通路, 使得  $I_{DDQ}$  急剧增大, 以至达到可检测的水平. 根据电路方程不难得到

$$I_{DDQ}^{f,t} = \frac{V_{DD}}{(R_{UP} + R_B) \parallel (R_{M_2} + R_{M_3}) + R_{ON}} \quad (1)$$

图 2 中所示为对该与非门电路的桥接阻值执行参数扫描分析得到的关系曲线  $I_{DDQ}(r)$ . 可见,  $I_{DDQ}$  与桥接阻值  $r$  成反比关系. 值得注意的是, 对不同的 RBF, 上述  $I_{DDQ}$  的表达式也不相同, 并由于晶体管的阻值非线性且与桥接阻值耦合, 难以用解析的方法求解 RBF 存在时的  $I_{DDQ}$ . 因此, 本文采用电路仿真的方案解决该问题.

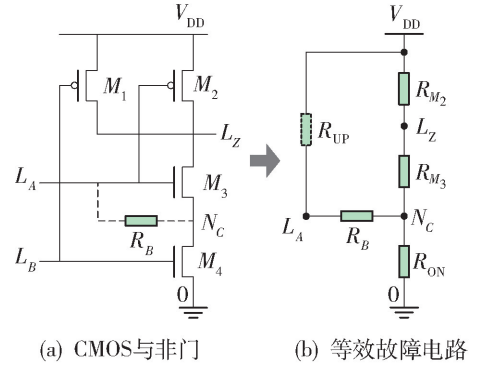


图 1 CMOS 与非门电路中的 RBF 示例

Fig. 1 RBF example of a CMOS nand-2 gate

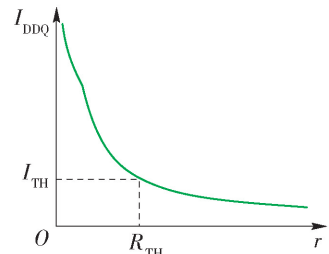


图 2  $I_{DDQ}$  与桥接电阻的关系

Fig. 2 Relationship between  $I_{DDQ}$  and bridging resistance

假设  $I_{DDQ}$  监控器的检测门限为  $I_{TH}$ , 其对应的临界桥接阻值为  $R_{TH}$ , 如图 2 所示, 则当  $I_{DDQ}^{f,t} \geq I_{TH}$  时, 监控器将产生告警, 表明故障被检测到. 因此, 对任意一个电阻式桥接故障  $f$ , 忽略系统噪声和测量误差的影响, 它被检测到的概率为

$$\rho_{f,t}(r) = \begin{cases} 0 & r > R_{TH} (I_{DDQ}^{f,t} < I_{TH}) \\ 1 & r \leq R_{TH} (I_{DDQ}^{f,t} \geq I_{TH}) \end{cases} \quad (2)$$

考虑桥接阻值的分布, 测试  $t$  对故障  $f$  的覆盖率定义为

$$FC_{f,t} = \frac{\int_0^{+\infty} \rho_{f,t}(r) \eta(r) dr}{\int_0^{+\infty} \eta(r) dr} = \frac{\int_0^{R_{TH}} \eta(r) dr}{\int_0^{+\infty} \eta(r) dr} \quad (3)$$

式中  $\eta(r)$  为桥接阻值分布的概率密度函数. 覆盖率  $FC_{f,t}$  表示在当前测试激励  $t$  下, 以  $I_{TH}$  为门限, 通过测量  $I_{DDQ}$  覆盖  $f$  全部阻值区间的概率. 进一步, 对于测试集  $T = \{t_1, t_2, \dots, t_n\}$ , 其对故障  $f$  的覆盖率定义为

$$FC_{f,T} = \max(FC_{f,t_1}, FC_{f,t_2}, \dots, FC_{f,t_n}) \quad (4)$$

式(4)表示如果测试集中有多个测试覆盖  $f$ , 则选取覆盖率最大的一个代表整体的故障检测能力.

最后, 定义测试集  $T$  对 RBF 故障集  $F = \{f_1, f_2, \dots, f_m\}$  的覆盖率为

$$FC_{F,T} = \frac{1}{m} \sum_{i=1}^m FC_{f_i,T} \quad (5)$$

式中  $m$  为 CUT 中 RBF 的数量.

## 2 RBF 仿真算法

RBF 仿真算法包括 2 个阶段: 电路级仿真和门级仿真. 电路级仿真的任务是基于 RBF 模型为基本逻辑门生成故障字典; 门级仿真则是通过在功能仿真过程中引入字典中记录的故障信息实现对测试集故障覆盖率的评估.

### 2.1 电路级仿真

电路级仿真的自动实现流程如图 3 所示. 给定目标门电路的描述, 首先将其解析为一种易于实现 RBF 注入的内部模型<sup>[8]</sup>, 随后故障注入模块根据给定的桥接节点和阻值生成原电路的故障版本. 仿真器调用 PSpice 仿真程序将故障电路置于不同的测试输入组合下进行仿真, 得到电路的  $I_{DDQ}$  测量值. 分析器将  $I_{DDQ}$  与  $I_{TH}$  进行比较, 若刚好满足  $I_{DDQ} = I_{TH}$ , 则令  $R_{TH} = R$ , 否则调整  $R$  的值并继续搜索  $R_{TH}$ . 最后, 将得到的  $R_{TH}$  保存在故障字典中.

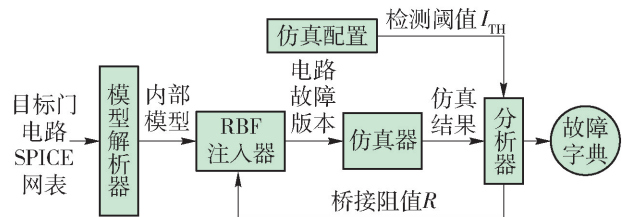


图 3 电路级 RBF 仿真过程

Fig. 3 RBF simulation at the circuit level

电路级仿真的关键步骤是搜索  $R_{TH}$ , 考虑到阻值搜索空间无上界(下界为 0), 这里采用动态边界折半搜索算法查找临界阻值  $R_{TH}$ , 算法流程如下.

#### 算法 1 动态边界折半搜索算法

1) 令  $R \approx 0$ , 执行一次理想 BF 仿真. 若  $I_{DDQ}^t <$

$I_{TH}$ , 说明无论  $R$  为何值  $t$  都不能检测  $f$ , 终止算法并返回  $R_{TH} = 0$ , 否则继续.

2) 令  $R = R_{INI}$  (初始值  $R_{INI} > 0$ ),  $R_L = R_H = 0$ , 执行一次 RBF 仿真. 若  $|I_{DDQ}^t - I_{TH}| \leq \varepsilon, 0 < \varepsilon \ll 1$ , 终止算法并返回  $R_{TH} = R$ , 否则继续.

3) 若  $I_{DDQ}^t > I_{TH}$ , 则令下限  $R_L = R$ , 若此时上限  $R_H \leq R$ , 则令  $R = 2 \times R$ , 否则  $R = (R + R_H)/2$ , 转到步骤 2).

4) 若  $I_{DDQ}^t < I_{TH}$ , 则令上限  $R_H = R$ , 并且  $R = (R + R_L)/2$ , 转到步骤 2).

值得注意的是, 在构建故障字典时, 应首先仿真一次无故障电路, 获取当前桥接节点  $n_1$  和  $n_2$  上的电压值, 只有当两节点电压值不等时才有必要启动故障仿真.

### 2.2 门级仿真

门级 RBF 仿真的实现过程如图 4 所示, 其输入为 CUT 的门级网表描述, 经过解析后转换为一种内部表示(包含构成 CUT 的逻辑门列表及其连接关系等信息). 仿真器利用测试生成器输出随机测试序列, 同时结合故障字典中记录的故障信息, 执行 CUT 的功能仿真和故障仿真. 最后按式(5)计算测试集  $T$  对 RBF 故障集  $F$  的覆盖率.

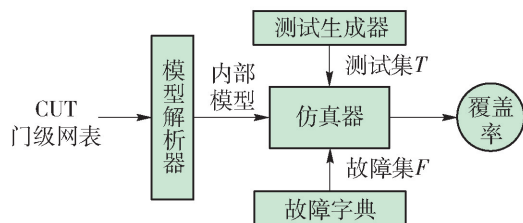


图 4 门级 RBF 仿真过程

Fig. 4 RBF simulation at the gate level

门级的故障仿真以正常的逻辑电路功能仿真为基础. 为此, 首先实现了一种事件驱动的组合逻辑电路功能仿真算法. 其主要思想是以 CUT 中信号值的变化为事件, 当一个门的输入信号上有事件发生时便调度这个门, 并利用其输出信号的变化产生新的事件, 以驱动后续仿真的运行, 直到 CUT 中不再产生新的事件为止. 每步仿真时刻被划分为若干无限小的  $\delta$ -周期, 每个  $\delta$ -周期处理当前的事件并调度相应的门, 并为下一个  $\delta$ -周期产生事件. 这样就能有效模拟实际逻辑电路中的器件并发行为<sup>[9]</sup>. 具体实现见算法 2.

#### 算法 2 逻辑电路功能仿真

1) 根据当前 CUT 原始输入端上的测试序列  $t$ ,

生成初始事件并赋给事件集  $E$ . 令即将被调度执行的门的集合  $P = \emptyset$ .

2) 若  $E = \emptyset$ , 则终止当步仿真, 否则根据  $E$  确定  $P$  中的元素. 设  $e \in E$ , 且与  $e$  关联的信号为  $s$ , 则将以  $s$  为输入信号的门作为  $P$  元素的候选, 再从中选择输入级数最小的门添加至  $P$  中. 连线的输入级数算法参见文献[10], 越靠近原始输入端的连线输入级数越小, 而门的输入级数则等于其输出连线的输入级数.

3) 令  $E = \emptyset$ , 调度执行  $P$  中所有的门, 并将由于其输出信号改变而产生的新事件添加至事件集  $E$ , 转到步骤2).

与基于逻辑测试的故障仿真算法不同, 基于  $I_{DDQ}$  测试的算法无需向 CUT 中实际注入逻辑故障, 而只需仿真一次无故障情形, 然后基于连线上的逻辑值查询故障字典得到  $R_{TH}$ , 再通过式(3)(4) 确定测试集对当前故障的覆盖率. 最终的测试集对由所有 RBF 构成的故障集的覆盖率由式(5)算得.

### 3 仿真实验及结果分析

首先为若干 CMOS 基本逻辑门构建了故障字典, 包括反相器, 缓存器, 两输入与门、或门、与非门、或非门和异或门. 表1中为给定  $I_{TH} = 100 \mu A$  时两

表1 两输入 CMOS 与非门的故障字典

Table 1 Fault dictionary of CMOS nand-2 gate k $\Omega$

RBF	"00"	"01"	"10"	"11"
$\langle L_Z, L_A \rangle$	32.50	26.25	0.00	0.00
$\langle L_Z, N_C \rangle$	0.00	27.50	0.00	0.00
$\langle L_Z, 0 \rangle$	45.00	38.75	38.75	0.00
$\langle L_Z, L_B \rangle$	32.50	0.00	27.50	5.00
$\langle L_Z, V_{DD} \rangle$	0.00	0.00	0.00	22.50
$\langle L_A, N_C \rangle$	0.00	0.00	0.00	31.88
$\langle L_A, 0 \rangle$	0.00	0.00	38.88	44.38
$\langle L_A, L_B \rangle$	0.00	32.50	33.75	0.00
$\langle L_A, V_{DD} \rangle$	38.75	43.75	0.00	0.00
$\langle N_C, 0 \rangle$	0.00	0.00	16.25	0.00
$\langle N_C, L_B \rangle$	0.00	22.50	5.00	27.50
$\langle N_C, V_{DD} \rangle$	0.00	38.75	0.00	38.75
$\langle L_B, 0 \rangle$	0.00	38.75	0.00	44.38
$\langle L_B, V_{DD} \rangle$	38.75	0.00	44.38	0.00
均值	13.39	19.20	14.61	15.31

输入与非门(如图1(a)所示)的故障字典实例. 其中第1列为该门电路中所有可能的14个RBF; 第2~5列为某一RBF分别在不同测试下对应的临界阻值  $R_{TH}$ . 晶体管仿真计算模型来自 Cadence 公司 OrCAD 16.3 设计软件中提供的器件模型库, NMOS 和 PMOS 管分别使用其默认的 MBreakN 和 MBreakP 模型.

注意到覆盖率与  $R_{TH}$  成正比, 因此要获得最大的故障覆盖率, 测试集应使逻辑门的输入对应故障字典中的最大临界阻值. 例如, 对于节点对  $\langle L_Z, L_A \rangle$ , 测试"00"的覆盖率高于其他测试的覆盖率, 因此, 应尽量使与非门的输入端出现"00"以使得  $\langle L_Z, L_A \rangle$  上的 RBF 能够获得最大的覆盖率. 然而, 若得到关于表1中所有 RBF 的最大覆盖率, 显然要求测试集能够遍历该与非门的所有输入组合. 若测试生成技术难以满足上述要求, 则应首先满足平均临界阻值最大的输入测试. 如表1中的"01", 其平均覆盖率好于其他测试.

由于实际电路中的桥接阻值  $R$  是分布于  $(0, +\infty)$  上的随机变量, 因此, 一种比较合理的假设是  $R$  服从对数正态分布

$$\eta(r) = \begin{cases} \frac{1}{\sigma r \sqrt{2\pi}} \exp\left(-\frac{(\ln r - \mu)^2}{2\sigma^2}\right) & r > 0 \\ 0 & r \leq 0 \end{cases} \quad (6)$$

式中:  $\sigma > 0$ ;  $\mu$  为实数. 在这一假设下式(3)可简化为

$$FC_{f,t} = \int_0^{R_{TH}} \eta(r) dr \quad (7)$$

实际应用中, 应根据历史故障信息或经验对桥接阻值的分布参数进行估计. 而作为仿真实验, 这里取  $\mu = 9$ ,  $\sigma = 1$ , 由此可得桥接阻值的期望值为  $\exp(\mu + \sigma^2/2) = 13.36 \text{ k}\Omega$ .

表2给出了针对 ISCAS'85 组合逻辑基准电路开展 RBF 仿真实验的结果. 其中, 第1列为 CUT 的名称; 第2列为 CUT 的规模, 即包含的基本逻辑门数量; 第3列为 CUT 中所有被仿真 RBF 的数量; 第4列为假定  $R = 0 \Omega$  时, 所有可能被  $I_{DDQ}$  测试检测到的 RBF 的数量; 第5列为基于传统 BF 覆盖率计算方法得到的覆盖率<sup>[6]</sup>, BF-FC 的值等于第4列数据与第3列的比; 第6列为假设桥接阻值取对数正态分布并根据本文所提方法计算得到的 RBF 覆盖率. 故障仿真实验中, c17 的测试集规模为  $V = 32$  (已穷举其所有输入组合), 其余基准电路采用随机生成的测试序列, 测试集规模均为  $V = 10^5$  (详见表3).



表2 组合基准电路 RBF 仿真结果 ( $I_{TH} = 100 \mu A$ )Table 2 RBF simulation results of the benchmarks ( $I_{TH} = 100 \mu A$ )

CUT	门数	RBF 数量	检测 数	BF-FC/ %	FC <sub>F,T</sub> / %
c17	6	84	84	100.00	91.23
c432	160	3 228	3 102	96.10	86.48
c499	202	5 880	5 672	96.46	83.22
c880	383	6 341	6 315	99.59	91.89
c1355	546	8 352	8 320	99.62	91.15
c1908	879	12 365	12 153	98.29	90.59
c2670	1 269	19 422	19 127	98.48	91.26
c3540	1 669	26 693	26 355	98.73	91.43
c5315	2 307	44 356	44 029	99.26	92.05
c6288	2 416	35 072	34 986	99.75	90.89
c7552	3 513	55 226	54 414	98.53	91.13

由表2可见,相比于传统的基于  $I_{DDQ}$  的理想 BF 测试,相同测试集对 RBF 的覆盖率明显偏小. 这说明传统方法存在覆盖率虚高的情况,而基于此生成的测试集将难以保证 CUT 中的真实桥接故障被有效地检测到,继而影响最终产品的可靠性.

表3 故障仿真效能比较

Table 3 Comparison with previous work

CUT	$V$	$B$	$K/s$	$K \cdot (BV)^{-1}/ms$	$V$	$B$	$K/s$	$K \cdot (BV)^{-1}/ms$
c432	$10^4$	5 253	1.24	$2 \times 10^{-5}$	$10^5$	3 228	5.59	$1 \times 10^{-5}$
c499	$10^4$	8 985	0.76	$1 \times 10^{-5}$	$10^5$	5 880	7.80	$1 \times 10^{-5}$
c880	$10^4$	$10^4$	3.70	$4 \times 10^{-5}$	$10^5$	6 341	16.54	$3 \times 10^{-5}$
c1355	$10^4$	$10^4$	7.17	$7 \times 10^{-5}$	$10^5$	8 352	26.60	$3 \times 10^{-5}$
c1908	$10^4$	$10^4$	2.97	$3 \times 10^{-5}$	$10^5$	12 365	52.05	$4 \times 10^{-5}$
c2670	$10^4$	$10^4$	4.69	$5 \times 10^{-5}$	$10^5$	19 422	83.79	$4 \times 10^{-5}$
c3540	$10^4$	$10^4$	3.18	$3 \times 10^{-5}$	$10^5$	26 693	120.90	$4 \times 10^{-5}$
c5315	$10^4$	$10^4$	1.34	$1 \times 10^{-5}$	$10^5$	44 356	492.43	$1 \times 10^{-4}$
c6288	$10^4$	$10^4$	13.77	$1 \times 10^{-4}$	$10^5$	35 072	235.87	$7 \times 10^{-5}$
c7552	$10^4$	$10^4$	3.29	$3 \times 10^{-5}$	$10^5$	55 226	621.06	$1 \times 10^{-4}$

由式(3)可知,影响最终覆盖率指标的因素有2个:桥接阻值分布  $\eta(r)$  和临界阻值  $R_{TH}$ . 图5所示为满足对数正态分布 ( $\mu = 9$ ) 的桥接阻值在不同的均方差  $\sigma$  下,故障覆盖率的变化情况,其中目标电路为 c432 基准电路. 可见,随着  $\sigma$  的增加覆盖率逐渐降低. 这是由于  $\sigma$  越小概率密度曲线收缩得越紧,通过积分求得的  $FC_{f_i}$  就越大(假设  $R_{TH}$  足够大),

表3中所列数据为将本文所提方法的仿真效能与文献[3]相比较所得结果. 其中,第1列为 CUT 名称;第2~5列为文献[3]中的数据;第6~9列为本文所得数据. 符号  $V$ 、 $B$ 、 $K$  分别表示测试用例数量、被仿真的 RBF 数量和仿真时间消耗.  $K/(BV)$  是归一化的仿真效能衡量指标,其值越小效能越高. 文献[3]的仿真实验环境为 Linux 操作系统、2.8 GHz AMD Opteron CPU 及 16 GB RAM. 本文中仿真实验的硬件配置为 Windows 7 操作系统、3.4 GHz Intel Core CPU 和 4 GB RAM,算法实现的软件环境为 Visual Studio 2010 C# 4.0. 通过比较可见,本文所提算法效能与文献[3]总体相当,两者效能指标相对起伏的主要原因是目标故障类型不同:文献[3]中主要研究门间 RBF,而本文针对门内 RBF,因此在故障仿真算法上存在差异. 另外值得注意的是,由于门间 RBF 需考虑全局节点的各种组合,其故障规模成几何级数增长,故对于较复杂电路,文献[3]中将故障数量限定为  $10^4$  (如表3中第3列的3~10行所示),而本文所研究的门内 RBF 规模与 CUT 中逻辑门数量成线性递增关系,因此通常情况下能够遍历所有故障.

最终求得的平均覆盖率也就越高. 然而,无论桥接阻值分布如何,RBF 覆盖率都不会超过传统的 BF. 这再次说明,传统方法过于理想化,结果偏乐观.

图6所示为覆盖率在不同的  $I_{TH}$  设定下的变化趋势(目标 CUT 仍为 c432). 显然,  $I_{TH}$  越小 ( $R_{TH}$  越大) 覆盖率越高. 并且基于 RBF 的覆盖率随着  $I_{TH}$  的变小逐渐趋于理想值 BF-FC. 然而,实际应用中并

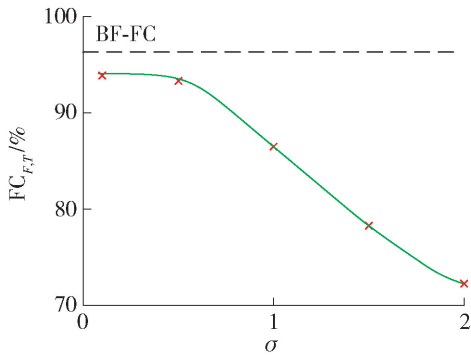
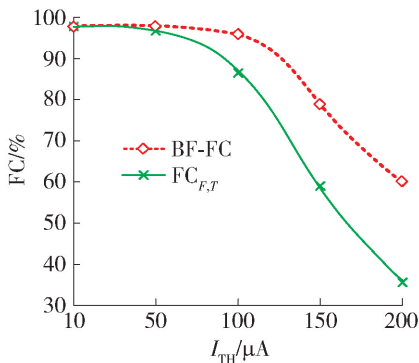


图5 桥接阻值分布对覆盖率的影响

Fig. 5 Relationship between  $FC_{F,T}$  and  $\eta(r)$ 

非  $I_{TH}$  越小越好。首先,要综合考虑高精度电流测量的实现难度和成本;其次,电流门限过小将使系统抗噪能力下降,造成虚警率增高,适得其反。本文所提方法为集成电路  $I_{DDQ}$  测试方案中  $I_{TH}$  的选取提供了参考。就本次仿真实验所采用的 CMOS 工艺、桥接阻值分布和目标电路 c432 而言,取  $I_{TH} = 50 \mu A$  即可获得较好的故障覆盖效果。

图6  $I_{TH}$ 对覆盖率的影响Fig. 6 Change of BF-FC and  $FC_{F,T}$  against  $I_{TH}$ 

## 4 结论

1) 在面向故障的集成电路测试中测试覆盖率受桥接阻值分布影响明显,传统的方法假设桥接阻值趋于零,得到的结果过于乐观。

2)  $I_{DDQ}$  测试中门限电流的设定十分关键,需综合权衡虚警、成本和故障覆盖率等要素,本文所提方法可为解决此问题提供一种参考。

3) 采用层次化的电路级和门级 2 级故障仿真算法,可显著提高仿真效能,增强了处理大规模的故障样本的能力,有助于提升测试集的质量。此外,门级的故障仿真结果又可作为更高电路设计抽象层次

(如寄存器传输级等)故障仿真的基础,从而进一步扩展方法的应用前景。

## 参考文献:

- [1] POLIAN I, ENGELKE P, BECKER B. Efficient bridging fault simulation of sequential circuits based on multi-valued logics [C] // Proceedings of IEEE International Symposium on Multiple-Valued Logic. Boston: IEEE, 2002: 216-222.
- [2] 马岚,王厚军. 基于输出电压和电源电流协同分析的故障诊断方法[J]. 仪器仪表学报, 2013, 34(8): 1872-1878.
- [3] MA L, WANG H J. Fault diagnosis method based on output voltage and supply current collaborative analysis [J]. Chinese Journal of Scientific Instrument, 2013, 34(8): 1872-1878. (in Chinese)
- [4] ENGELKE P, BECKER B, RENOVELL M, et al. SUPERB: simulator utilizing parallel evaluation of resistive bridges [J]. ACM Transactions on Design Automation of Electronic Systems, 2009, 14(4): 56.
- [5] ZHONG S, KHURSHED S, AL-HASHIMI B M. A fast and accurate process variation-aware modeling technique for resistive bridge defects [J]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2011, 30(11): 1719-1730.
- [6] AZEVEDO J, VIRAZEL A, BOSIO A, et al. Impact of resistive-bridge defects in TAS-MRAM architectures [C] // Proceedings of IEEE Asian Test Symposium (ATS). Niigata: IEEE, 2012: 125-130.
- [7] CHAKRAVARTY S, THADIKARAN P J. Simulation and generation of IDDQ tests for bridging faults in combinational circuits [J]. IEEE Transactions on Computers, 1996, 45(10): 1131-1140.
- [8] RAJSUMAN R. Iddq testing for CMOS VLSI [J]. Proceedings of the IEEE, 2000, 88(4): 544-568.
- [9] XU A Q, TANG X F, YANG Z Y. Mutant generation for analog circuit designs [C] // Proceedings of IEEE International Conference on Software Engineering and Service Sciences. Beijing: IEEE, 2014: 916-920.
- [10] 李晓维,吕涛,李华伟,等. 数字集成电路设计验证——量化评估、激励生成、形式化验证[M]. 北京: 科学出版社, 2010: 12-18.
- [11] 王新安,蒋安平,宋春殫,等. 数字系统测试[M]. 北京: 电子工业出版社, 2007: 48-49.

(责任编辑 吕小红)