

应用 Variable-Tail 编码压缩的测试资源划分方法

韩银和, 李晓维, 徐勇军, 李华伟

(中国科学院计算技术研究所, 北京 100080; 中国科学院研究生院, 北京 100039)

摘 要: 测试资源划分是降低测试成本的一种有效方法. 本文提出了一种新的有效的对测试数据进行压缩的编码: Variable-Tail 编码, 并构建了基于该编码的测试资源划分方案. 文章的理论分析和实验研究表明了采用 Variable-Tail 编码能取得比 Golomb 编码更高的压缩率, 针对多种模式下的测试向量均能提供很好的压缩效果, 解码器的硬件也较易实现. 文章还提出了一种整合不确定定位动态赋值的测试向量排序算法, 该算法可以进一步提高测试压缩率. 文章最后用实验数据验证了所提编码和排序算法的高效性.

关键词: Variable-Tail 编码; Golomb 编码; 可适应性; 测试模式; 诊断模式

中图分类号: TP391.76 **文献标识码:** A **文章编号:** 0372-2112 (2004) 08-1346-05

Test Resource Partitioning Using Variable-Tail Code

HAN Yin-he, LI Xiao-wei, XU Yong-jun, LI Hua-wei

(Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100080, China;

Graduate School of Chinese Academy of Sciences, Beijing 100039, China)

Abstract: Test resource partitioning is an efficient method to reduce the test cost. This paper presents a novel and efficient code, i. e. variable-tail code, for test data compression. Theoretical analysis and experimental study show that the variable-tail code can provide the higher compression efficiency than Golomb. It is suitable for different test sets and could provide the high compression ratio. The decoder of the variable-tail code is simple and easy to be implemented. In order to achieve higher compression ratio, an efficient test vectors reordering algorithm (ERA) incorporating a dynamic X-bit assignment procedure is presented. Experimental results demonstrated the efficiency of the proposed code and algorithm.

Key words: Variable-Tail code; Golomb code; flexibility; test mode; diagnose mode

1 引言

随着基于 IP Core 的 SOC 设计的普及, 单个 SOC 设计上集成的 IP Core (比如, 微处理器, 存储器, DSPs, I/O 控制器等) 越来越多. 大量 IP Core 被集成到一个芯片上, 使得对该芯片进行有效的测试成为一个难题. 通常, 对应着每一个 IP Core, 其生产厂商会提供对应的测试数据. 芯片上集成的 IP Core 越多, 那么在实际测试时候需要的测试向量数据也越大. 这就不但对测试设备 (ATE) 的存储性能要求越来越高, 同时也增加了实际租用测试设备的时间, 使得测试成本越来越高.

解决测试数据量的一个方法是采用基于测试数据压缩/解压缩的测试资源划分^[1-9]. 测试压缩分为两个方面: 测试激励压缩和测试响应压缩. 在基于激励压缩方法中, 首先采用特定的算法 (Run-length^[1], Golomb^[2]和 FDR^[3], SAC^[5], 本文提出的 Variable-Tail 编码) 对由 IP Core 厂商提供的测试数据进行压缩, 然后将压缩后数据存放到工作站上. 在测试之前, 将存

放在工作站上的压缩数据上载到测试设备主存中. 在测试时, 测试通道将存放在主存中的数据加到芯片的引脚上, 通过芯片内的解压缩电路, 将数据解码还原成原始测试向量, 并将原始测试向量加到对应的 IP Core 上进行测试.

为了进一步提高压缩比率, 减少测试时间, 本文提出了新的编码压缩算法: Variable-Tail 编码. 文章首先描述了 Variable-Tail 编码的构成 (第 2 节), 并从理论上分析了使用 Variable-Tail 编码的压缩率和测试时间 (第 3 节). 接着第 4 节提出了一种新的测试向量排序算法, 并提出了对不确定定位进行动态赋值策略. 第 5 节构建了解码器框架. 文章最后用实验数据证明了 Variable-Tail 编码是一种压缩效果好, 易于实现的, 适应性很强, 测试应用时间少的编码压缩算法.

2 Variable-Tail 编码

Variable-Tail 编码是一种前缀编码, 它提供了变长-变长编码方式. 在编码方式和 Golomb^[2]有相似之处. 与 Golomb 编

收稿日期: 2003-09-15; 修回日期: 2004-02-27

基金项目: 国家自然科学基金 (No. 90207002; No. 602420010); 北京市重点科技项目 (No. H020120120130); 中科院计算所基础研究基金 (No. 20036160)

码不同的是:Variable-Tail 编码中,组与组编码的尾码长度是不相同的,依次增加一位.其中一组(编码前缀相同的游程组成)所包含游程个数总是其上一组所包含游程个数的两倍.我们定义在 Variable-Tail 编码中,基数 r 为第一组编码的尾码长度.下图列出 $r=2$ 时的 Variable-Tail 编码表:

Variable-tail 编码 ($r=2$)				
游程	组号	组前缀	组尾码	编码
0			00	000
1	1	0	01	001
2			10	010
3			11	011
4			000	10000
5			001	10001
6			010	10010
7	2	10	011	10011
8			100	10100
9			101	10101
10			110	10110
11			111	10111
12	3	110	0000	1100000
13			0001	1100001
..

图 1 $r=2$ 时,Variable-Tail 编码表

Variable-Tail 编码对测试流程的适应性使得其对测试模式下的测试向量和诊断模式下的测试向量均能提供很好压缩效果,下面的分析将帮助理解这一点.

3 Variable-Tail 编码分析

3.1 压缩效果分析

假设我们压缩的对象是 T_{diff} , Variable-Tail 编码采用基数为 r ,则有:

引理 在 Variable-Tail 编码中第 k 组所代表的游程范围为: $L_E = 2^{\lceil \log_2(I + 2^r + 1) - r \rceil} + r - 1$

对于游程为 l 的数据对应的编码长度为:

$$2^{k+r-1} - 2^r - 1 < l_k \leq 2^{k+r} - 2^r - 1$$

证明从略

可以用概率论的方法计算压缩效果.假设测试数据中某一位置 0 的概率为 p ,置 1 的概率则为 $(1-p)$,其中 $0 < p < 1$,长度为 l 的游程出现的概率就应该为: $p^l(1-p)$.那么原始数据中平均游程长度:

$$\bar{D} = 1 + \sum_{l=1}^{\infty} l p^l (1-p) = \frac{1}{1-p}$$

从上面引理我们得到对于组 k ,其代表的最短和最长游程分别为 $2^{k+r-1} - 2^r$ 和 $2^{k+r} - 2^r - 1$,所以对于任意给定的一个游程 l ,它属于组 k 的概率由下式给出:

$$P(l, k) = \frac{(2^{k+r} - 2^r - 1)}{2^{(k+r-1)} - 2^r} p^l (1-p) = \frac{1}{p^2} (p^{2^{k+r-1}} - p^{2^{k+r}})$$

而在组 k 中所包含的每一个游程对应的编码长度都是相等的,且为: $L_k = 2k + r - 1$,因此对组 k 及其出现的概率进行加权就可以得出使用 Variable-Tail 平均编码长度,即:

$$\begin{aligned} \bar{V} &= \sum_{k=1}^{\infty} P(l, k) * L_k = \sum_{k=1}^{\infty} \frac{1}{p^2} (p^{2^{k+r-1}} - p^{2^{k+r}}) * (2k + r - 1) \\ &= \frac{2}{p^2} \sum_{k=1}^{\infty} (p^{2^{k+r-1}}) + (r - 1) \end{aligned}$$

我们用压缩倍数: $V = \frac{\bar{D}}{\bar{V}}$ (编码前数据和编码后数据总位数之比) 来表征压缩效果:

$$v = \frac{\bar{D}}{\bar{V}} = \frac{1}{(1-p) \left(\frac{2}{p^2} \sum_{k=1}^{\infty} p^{2^{k+r-1}} + r - 1 \right)} \quad (1)$$

上式没法得到进一步的结果,但是,由于指数关系存在,所以我们只要用 $k=1$ 到 1000 来对上式进行估算能够得到足够的精度.我们可以用同样的方法来分析 Golomb 编码,得出:

$$g = \frac{\bar{D}}{G} = \frac{1}{(1-p) (\log_2 m + 1 / (1 - p^m))} \quad (2)$$

图 2 比较了 0.90 $\leq p < 1.00$ 情况下,由上两式决定的压缩倍数,图中 Golomb 编码参数 m 取文献 [2] 实验证明通常有很高压缩率的 $m=4$, Variable-Tail 编码也取我们在实

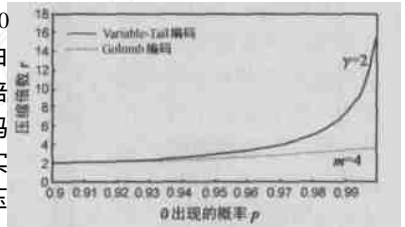


图 2 两种编码的理论分析结果

验中使用比较多的 $r=2$ 来进行比较.从上图可以看出,当 $p > 0.92$ 时,使用 Variable-Tail ($r=2$) 编码的压缩倍数总是大于使用 Golomb ($m=4$) 编码的压缩倍数.在原始测试数据进行优化和差分后, p 一般都在 0.93 以上,从理论分析结果可以看出使用 Variable-Tail 编码压缩效果要优于使用 Golomb 编码的.

3.2 测试时间分析

下面我们来分析利用 Variable-Tail 编码进行测试所用时间上限,假设自动测试设备 (ATE) 的测试频率为 f_{scan} ,芯片扫描链的工作频率为: f_{ATE} ,既然 ATE 和扫描链工作在不同频率,解码器就必须分成两个工作频率不同的部分,一部分工作频率为 f_{scan} ,另一部分工作频率为 f_{ATE} .我们引入一个比率 α ,定义 $\alpha = f_{scan} / f_{ATE}$.用 T 表示总测试时间, T_{shift} 定义为将编码从测试仪输出到解码器的时间, T_{decode} 是解码器解码时间.很容易得到:

$$T_{shift} = \frac{|L_E|}{f_{ATE}}, T_{decode} = \frac{|L_{Diff}|}{f_{scan}} \quad (|L_{Diff}|, |L_E| \text{ 分别为差分数据及其编码的位数})$$

如果假设解码器在工作之前,要将所有数据都送给解码器,我们可以将这时总解码时间作为我们解压测试时间上限,因为实际移入编码和解码过程是并行工作的,解码时间要小于这个假设时状态下的时间:

$$T \leq T_{shift} + T_{decode} = \frac{1}{f_{ATE}} (|L_E| + \frac{|L_{Diff}|}{\alpha})$$

在下面的实验中我们要用到这个公式来说明即使是这个上限时间,也比采用 ATPG 的压缩方法所用的测试时间少.

4 有效的测试向量排序算法

观察公式(1),如果给定一个测试向量和决定使用 Variable-Tail 编码,提高压缩比率的一个方法就是提高 p . p 的取值和测试向量及其排列顺序相关. 我们可以通过对测试向量的顺序进行重排序来提高测试向量的压缩率. 因为我们对差分数据进行压缩,所以提高 p 就可以等效于减少差分向量上的海明距离. 对于一对向量 t_i/t_{i+1} (每个向量有 m 位),海明距离定义为: $HD(t_i, t_{i+1}) = \sum_{j=1}^m [t_i(j) \oplus t_{i+1}(j)]$.

我们可以用一个无向图来刻画测试向量海明距离的问题. 在这个图中,每一个节点代表一个向量,相邻节点之间的权值为两节点所代表向量之间的海明距离. 我们称这个图为 HD-图. 图 3 左侧是 C17 电路的测试向量,右侧是为 C17 向量构造的 HD-图.

要想提高压缩比率,就要尽量减少待测向量上的海明距离. 而降低海明距离在 HD-图上等效于找一个具有较小权值的哈密尔顿

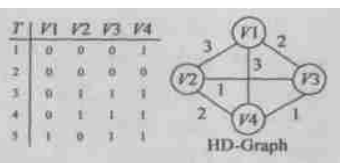


图 3 C17 的测试向量及 HD-图

路径. 以图 3 为例,如果测试向量顺序以 $O(V_1, V_2, V_3, V_4)$ 为序排列,那么利用 Variable-Tail 编码进行压缩后为 18 位. 但是如果测试向量排列顺序以最小权值路径 $O(V_1, V_3, V_2, V_4)$ 为序的话,利用 Variable-Tail 编码进行压缩,可使得编码后位数减少到 16 位.

对于一个类似于图 3 中的无向图寻找一个哈密尔顿路径是一个 NP 问题. 不可能有一个在多项式时间复杂度内的解法. 大多数方法都集中于使用一个特定目标的贪婪算法去解决这类排序算法. 本文提出一个有效的海明距离排序算法,该算法首先将所有向量划分为两个集合,已排序集合和待排序集合. 初始化的时候,已排序集合为空,然后向其中放入第一个向量,以后要放入已排序集合的向量必须保证其和最后一个已排序中的向量的海明距离在一个期望值以内. 如果超过这个期望值,那么就要回溯,将已排序中最后一个向量删除,重新选择新的向量放入其中,如果还不能满足要求,则继续向上一级回溯. 整个回溯是通过一个期望的最大海明距离和最大回溯深度来控制的. 下面是我们提出的有效海明距离排序 (ERA) 算法的伪代码:

```

Algorithm ERA
S :set of ordered vectors(empty initially)
Add vector 1 to S
While(all vectors not covered)
  Find a vector V uncovered whose HD is
  minimal to the latest vector of S
  if ( HD < A Limited Value HDmax)
    Add V to S
    Assign the X bit of V to minimal HD
  else
    Delete the latest vector of S
    if(not exceed max backtracking depth dmax)
      Backtrack
    endif
End while

```

图 4 ERA 伪代码描述

在这个算法中,有两个手工设定的参数:相邻向量最大海明距离 HD_{max} 和最大回溯深度 dmax. HD_{max} 是用来激发回溯的. 这两个参数构成了算法复杂度和测试向量压缩率的一个折衷. 如果期望测试向量总海明距离近可能的小,那么可以设较小的 HD_{max} 和较大的 dmax,产生这样的优化结果,算法运行时间会很长. 如果用户期望以较小的时间产生新的向量顺序,那么可以选取较大的 HD_{max} 和较小的 dmax.

文献[10]说明了采用不同策略对不确定位赋值对于压缩率有很大的影响. 对不确定位进行赋值有很多方法. 文献[10]采用的不确定位赋值算法是基于 ATPG 的,这种方法虽然可以得到较大的压缩率,但是不适合用于把核作为黑盒的 SOC 测试. 在上述我们排序算法中,采用了动态赋值而非初始化的时候静态赋值,这一点将很大程度上提高压缩率. 因为如果有回溯,那么不确定位就要重新赋值,这样算法总可以保证一个合理的赋值使得现有顺序下海明距离比较小.

5 解码器结构

Variable-Tail 编码的解码器可用一个有限状态机和三个计数器来实现,具体各信号之间的时序关系及状态迁移关系可参考文章^[3],解码器的操作流程描述如下:

(1) 有限状态 (FSM) 首先接受前缀. 在向 k -计数器移入第一个 1 之前,首先利用 r -计数器,向 k -计数器移入 r 个 0,然后再将前缀移入 k -计数器 ($shift, en = 1, inc1$ 也为 1, $\log_2 k$ -计数器移入前缀的位数,当接收到 0 时,计数、移位停止 ($shift = 0, inc1 = 0, en = 0$).

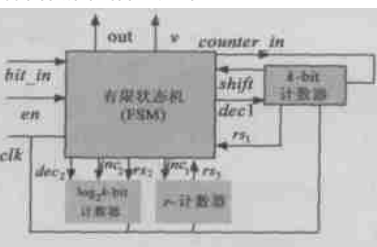


图 5 解码器结构框图

(2) k -计数器执行减法操作, out 逐个输出 $0 (v = 1)$, 直至 n_1 为 1 为止.

(3) 联合利用 $\log_2 k$ -计数器和 r -计数器,将后缀移入 k -计数器 ($dec_2 = 1, inc_2 = 1$),直到 n_2, n_3 均为 1.

(4) 再对 k -计数器执行减法操作,同时 out 逐个输入 $0 (v = 1, en = 0)$,直至 $n_1 = 1$ 为止,并在最后一位输出个 1,本次编码解码结束.

需要特殊说明的是,图 4 解码电路适合于基数 $r \geq 1$ 的情况. 如果 $r = 0$,图 4 的解码电路需要进行适当的改变. 图 4 中的 r -计数器将不再需要,同时,上述解码过程第 3 步,当 $\log_2 k$ -计数器被减少到 1 (而非 0) 时, n_2, n_3 为 1.

6 实验结果

下面用实验来评估我们的压缩算法. 实验使用的 PC 机采用 Intel P4 1.6G CPU, 128M 内存,工作站采用 SUN Ultra 60, 我们的算法是在 PC 机上完成的,部分测试向量数据是在工作站上产生的.

首先用实验来说明正常测试模式下,对同样的测试数据,

使用 Variable-Tail 编码压缩效果比 Golomb 和 FDR 等编码要好.我们针对 ISCAS89 中几个比较大的时序电路,采用美国 Duke 大学提供的 MinTest^[11]产生的测试向量集(和文献[2~4]采用的测试向量数据是一致的).在应用 Variable-Tail 编码压缩之前,首先对目标向量集进行差分处理并应用 ERA 排序.表 1 中列出各编码的最好压缩率,及对应于最高压缩率时的参数值设置.从表 1 可以看出 Variable-Tail 编码大大提高了压缩率,跟以前的编码相比,使用 Variable-Tail 编码后的数据量仅相当于采用其他编码数据量的 70%.由于我们采用 ERA 算法,所以即使同样使用 FDR 编码(当 $r=1$ 时,Variable-Tail 编码等效于 FDR 编码),我们的压缩结果也比文[3]中要好,如表中的 S5378, S15850, S38584.同时,Variable-Tail 编码比采用 Alt Run-length 压缩率要高,采用 Alt Run-length 还要带来额外的硬件开销,而采用 Variable-Tail 则没有这方面的问题.

表 1 正常测试模式下测试向量压缩(测试向量全部由 Mintest 产生,由于文献[2]中 S35932 是由 Atalanta^[9]产生的,压缩结果没有可比性,故没列出)

电路	原始数据 据位数	Golomb ^[2]		FDR ^[3]	Alt Run-length ^[4]	Variable-Tail 编码	
		m	编码位数			r	编码位数
S5378	23754	4	14085	12306	—	1	9420
S9234	39273	4	22250	21644	21612	2	14636
S13207	165200	16	41658	35226	32648	2	21081
S15850	76986	4	40717	36276	26306	1	20969
S35932	28208	—	—	25332	—	0	11632
S38417	164736	4	92054	74896	64976	0	56495
S38584	199104	4	104111	93860	77372	1	69964

为了验证针对不同模式测试向量,Variable-Tail 编码的有效性.我们使用 ATOM^[12]针对 ISCAS 85 中组合电路产生诊断模式的向量.从表 2 可以看出,在诊断模式下,测试数据比在正常测试模式下产生的测试数据量多得多,相对于正常测试模式下,诊断模式压缩率也会更高.表 2 列出了使用 Golomb, FDR 及 Variable-Tail 编码能取得的最大压缩率下对应的参数值及编码位数.可以看出,使用 Golomb 编码取得最大压缩效果的时候, m 变动比较大.而使用 Variable-Tail 不仅压缩率更高,而且参数 r 变化幅度很小,这说明了 Variable-Tail 编码的适应性很强,针对不同的向量均能取得较好的压缩效果.

表 2 诊断模式下测试向量压缩

电路	原始数据 位数	Golomb 编码		FDR 编码 位数	Variable-Tail 编码	
		m	编码位数		r	编码位数
C1355	64206	64	7068	6432	2	6171
C1908	61710	32	8654	8770	3	8080
C2670	612790	256	15902	17851	5	15667
C3540	164550	64	18412	20029	4	17998
C5315	941798	256	31040	38167	6	30419
C6288	246720	16	55877	48890	2	47797
C7552	1535733	256	48797	57197	6	48103

接下来我们用实验比较使用 Variable-Tail 编码压缩和使用基于 ATPG 动态压缩测试向量集的测试时间.从表 3 可以看出,即使对于应用 Variable-Tail 编码的测试时间上限(计算方法参考第 3 节),也都小于应用 MinTest(被认为是到目前为止

止最好的 ATPG 压缩工具)测试向量的测试时间.由此看来,相对于 ATPG 压缩算法而言,采用 Variable-Tail 对测试数据进行编码可以缩短测试时间,降低测试成本.

表 3 采用 Variable-Tail 压缩的测试时间和采用 MinTest 压缩测试时间比较

电路	测试设备 工作频率		采用 Variable-Tail 编码 测试时间上界 T (ms)	使用 MinTest 测试向量 测试时间 T (ms)
S5378	50	4	0.3072	0.4152
S9234	50	4	0.4891	0.5187
S13207	50	4	1.2476	3.2620
S15850	50	4	0.8043	1.1487
S35932	50	4	0.3737	0.3879
S38417	50	4	1.9536	2.2630
S38584	50	4	2.3948	3.2208

7 结语

近来的研究表明,基于压缩/解压缩的测试资源划分,是测试资源划分策略的一个重要发展方向.本文提出了一种新的有效的编码-Variable-Tail 编码,由于该编码具有高压压缩率高(相对以前的编码提高 30%)和很好的可适应性,使得它比 Golomb, FDR 等编码更适用于低成本测试资源划分策略中.本文还提出了一种有效的测试向量排序算法可以进一步提高测试压缩率.结合 Variable-Tail 编码和排序算法,可以取得几乎接近于理想编码压缩的最大压缩率的效果.从而大大减少待测芯片的测试成本.

致谢 本文作者要感谢 Synopsys 公司的 Anshuman Chandra 博士,他能够在繁忙的工作之余,审阅了本文提出的编码和算法,并且对于我们的实验提出了一些改进的建议.同时还要感谢 Duke 大学的 K. Chakrabarty 教授和 Li Lei 博士生,为我们提供了他们实验用的测试向量和实验结果.

参考文献:

- [1] A Jas J, Ghosh-Dastidar, N A Toubia. Scan vector compression/ decompression using statistical coding[A]. Proceeding of 17th IEEE VLSI Test Symposium[C]. Dana Point, California, USA, 1999. 114 - 120.
- [2] A Chandra, K Chakrabarty. System-on-a-Chip test data compression and decompression architectures based on Golomb codes[J]. IEEE Trans. on CAD of Integrated Circuits and System, 2001, 20(3): 355 - 368.
- [3] A Chandra, K Chakrabarty. Frequency-directed run length (FDR) codes with application to system-on-a-chip test data compression[A]. Proceeding of 20th IEEE VLSI Test Symposium[C]. Marina Del Rey, California, USA, 2001. 42 - 47.
- [4] A Chandra, K Chakrabarty. Reduction of SOC test data volume, scan power and testing time using alternating run-length codes[A]. Proceeding of IEEE/ACM, Design Automation Conference[C]. New Orleans, Louisiana, USA, 2002. 673 - 678.
- [5] 徐磊,孙义和.基于 SOC 架构的可测性设计方法学研究[D].北京:清华大学研究生院,2002.
- [6] A Chandra, K Chakrabarty. How effective are compression codes for reducing test data volume [A]? Proceeding of VLSI Test Symposium

- [C]. Monterey, California, USA, 2002. 91 - 96.
- [7] L Li, K Chakrabarty. Test data compression using dictionaries and fixed-length indices [A]. Proceeding of IEEE VLSI Test Symposium [C]. Napa Valley, California, USA, 2003. 219 - 224.
- [8] Yinhe Han, Yongjun Xu, Xiaowei Li. Co-optimization for test data compression and testing power based On variable-tail code[A]. Proceeding of 5th International Conference on ASIC[C]. Beijing, P R China, 2003. 105 - 108.
- [9] Yinhe Han, Yongjun Xu, Huawei Li, Xiaowei Li, A. Chandra. test resource partitioning based on efficient response compaction for test time and tester channels reduction [A]. Proceeding of Asian Test Symposium[C]. Xi'an, ShanXi, P R China, 2003. 440 - 445.
- [10] K Miyase, S Kajihara, I Pomeranz, M Reddy. Don't-care identification on specific bits of test patterns[A]. Proceeding of International Conference on Computer Design[C]. Freiburg, im Breisgau, Germany, 2002. 194 - 199.
- [11] I Hamzaoglu, J H Patel. Test set compaction algorithms for combinational circuits[A]. Proceeding of International Conference on CAD[C]. San Jose, California, USA, 1998. 283 - 289.

- [12] I Hamzaoglu, J H Patel. New techniques for deterministic test pattern generation[A]. Proceeding of VLSI Test Symposium [C]. Princeton, New Jersey, USA, 1998. 446 - 452.

作者简介:



韩银和 男, 1980 年生于江苏省阜宁县, 博士研究生, 1997 年考入南京航空航天大学, 2001 年获得南京航空航天大学工学学士学位, 并于同年进入中国科学院计算技术研究所攻读博士学位, 主要研究方向是: SOC 测试资源划分和优化, IC 可测性及测试技术, ATE 设计技术。



李晓维 男, 1964 年出生, 博士, 研究员, 博士生导师, IEEE 高级会员, 研究方向: VLSI/ SoC 低功耗易测试设计、设计验证与测试、可信计算。

www.cnki.net