

基于 TSV 绑定的三维芯片测试优化策略

神克乐¹, 虞志刚¹, 白宇²

(1. 清华大学计算机系, 北京 100084; 2. 清华大学软件学院, 北京 100084)

摘要: 本文提出一种三维片上系统(3D SoC)的测试策略, 针对硅通孔(TSV, Through Silicon Vias)互连技术的3D SoC 绑定中和绑定后的测试进行优化, 由于测试时间和用于测试的 TSV 数目都会对最终的测试成本产生很大的影响, 本文的优化策略在有效降低测试时间的同时, 还可以控制测试用的 TSV 数目, 从而降低了测试成本. 实验结果表明, 本文的测试优化策略与同类仅考虑降低测试时间的策略相比, 可以进一步降低约 20% 的测试成本.

关键词: SoC 测试; 3D SoC; 测试优化; 测试成本

中图分类号: TP391.76; TN407 **文献标识码:** A **文章编号:** 0372-2112 (2016)01-0155-05

电子学报 URL: <http://www.ejournal.org.cn>

DOI: 10.3969/j.issn.0372-2112.2016.01.023

Optimization Strategy for TSV-Based 3D SoC Testing

SHEN Ke-le¹, YU Zhi-gang¹, BAI Yu²

(1. Dept. of Computer Science, Tsinghua University, Beijing 100084, China;

2. School of Software, Tsinghua University, Beijing 100084, China)

Abstract: The optimization problem of three dimensional system on chip (SoC) needs to be solved before it enters the market. We propose a reconfigured test architecture optimization of TSV-based (Through Silicon Vias-based) 3D SoC, and the optimization includes both mid-bond testing and post-bond testing. As both test time and the number of TSV for test impact the overall test cost, our proposed scheme can reduce overall test time, while controlling the number of TSVs. Experiment results show that our scheme achieves around 20% on the reduction of test cost compared with one baseline solution which only considers reducing test time.

Key words: SoC testing; 3D SoC; optimization; test cost

1 引言

近些年随着片上系统(SoC, System on Chip)的发展越来越快, 片上系统已经到了一个性能的瓶颈, 主要是归结于电路的延迟越来越大, 功耗越来越高. 为了延续摩尔定律^[1], 三维(3D, 3 Dimensional)集成技术日趋成为了一个很有前景的解决方案. 首先, 三维集成技术可以通过利用硅通孔(TSV, Through Silicon Via)互连来降低芯片的连线长度, 从而降低电路延迟^[2]; 其次, 由于连线长度的降低, 芯片的功耗也下降了; 最后, 三维集成支持混合技术(mixed-technology)的实现, 即不同层可以放置不同的工艺包括微电子机械系统、图像传感器等等, 这样更有利于混合工艺的片上系统的发展^[4].

三维片上系统(3D SoC)从连接关系分, 可以有以下几种: 面对面连接(face to face), 背对背连接(back to

back)和面对背连接(face to back)^[3]. 所谓“面”, 就是三维片上系统每一层金属的那一面, 而“背”就是电路的衬底, 由于只有面对背支持超过两层以上的三维集成, 所以面对背连接是运用最广泛的.

三维集成电路从每一层晶片的实现方式来分有以下三种: (1)硬晶片(hard die): 测试结构已经存在; (2)软晶片(soft die): 需要自己设计测试结构; (3)固晶片(firm die), 测试结构部分已设计好.

虽然三维集成技术前景很好, 但是, 在这项技术成熟前, 科学家和工程师们还需要应对很多问题和挑战, 而其中对三维集成芯片的测试, 被有些专家认为是“第一挑战”^[5]. 衡量测试好坏的重要因素就是测试成本^[6], 而影响测试成本最重要的因素就是测试时间, 因此, 设计出一种优化的测试结构, 降低测试时间, 便可以降低测试成本, 从而可以更早让这项技术趋于成熟,

进入市场.

针对 3D SoC 的测试,业内广泛运用的就是模块化测试(modular testing)^[7].这种测试策略是将 SoC 里面的核(core)通过测试外壳(test wrapper)与周围的逻辑电路隔离,然后运用测试访问机制(TAM, Test Access Mechanism)从核的输入输出端口传递测试激励,并且将测试响应传输给测试仪(ATE).而 3D SoC 的测试,比传统的二维 SoC 测试更加复杂,首先是因为测试访问机制需要穿越多层来传输测试激励及测试响应数据,因此必须有专门用于测试的 TSV;其次,3D SoC 有着复杂的测试流程,文献[8]提出了广泛应用的 3D SoC 测试流程,包括了绑定前,绑定中和绑定后测试.

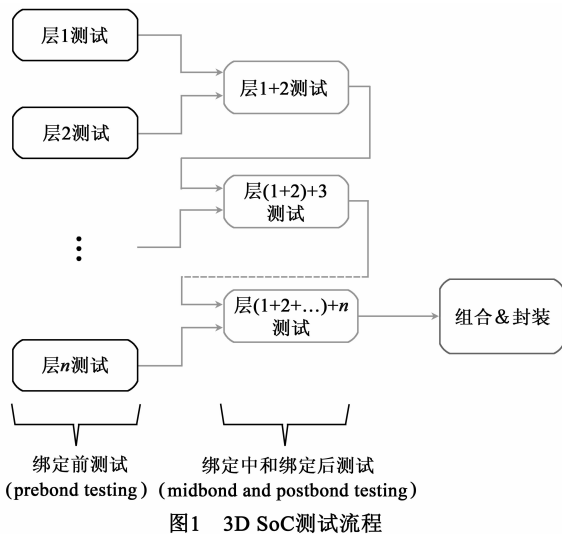


图1 3D SoC测试流程

图1展示了包含 n 层 3D SoC 的测试流程.第一列表示的就是绑定前测试(pre-bond testing),即每层在绑定前都要单独测试,第二列中的“层1+2测试”到“层1+2+...+ $n-1$ 测试”都是绑定中测试(mid-bond testing),这是因为 3D SoC 在生产过程中,每一次新堆叠一层,都要测试一遍,用来保证堆叠过程中层与层之间的连接是无故障的,如果有故障,可以提前告知这个芯片是故障芯片,从而降低测试时间^[13].第二列最后一行即所有层绑定之后的测试称为绑定后测试(post-bond testing).

二维 SoC 的测试优化课题已经有很多很好的解决方案^[9,10].然而,由于 3D SoC 的复杂性,传统的方案不能适用在新的 3D SoC 上,因此,设计一种新的 3D SoC 的测试优化策略显得尤为重要.文献[11]提出了一种针对 TSV 绑定后的测试优化策略,通过这种策略,测试时间得到了降低,同时 TSV 的数目也在可控范围内,然而,这种策略并没有考虑绑定中的测试优化.文献[12]通过改变三维芯片的堆叠次序来降低测试时间,虽然

该文献考虑了绑定中的测试优化,但是却并没有考虑改变堆叠次序会带来 TSV 数目的提高,而 TSV 的数目,也会影响最终的测试成本.

因此,本文设计了一种新颖的 3D SoC 硬晶片(hard die)测试策略,主要贡献包括:

- (1)建立了总测试成本模型,同时考虑测试时间和用于测试的 TSV 数目对成本的影响;
- (2)通过改变 3D SoC 的芯片堆叠次序,针对绑定前测试和绑定后测试都进行了优化,更好的降低了测试成本.

本文的组织形式如下,第2节给出了本文提出的方法,首先对测试优化问题进行定义,然后详细阐述了本文提出的测试调度策略.所提的 3D-SoC 测试优化策略评估与分析在第3节中展示.最后第4节总结全文.

2 所提的 3D-SoC 测试优化策略

2.1 3D-SoC 的测试模型

首先,本文给出 3D SoC 测试成本模型:

$$\text{Cost} = \alpha * t + (1 - \alpha) * \text{tsv} \quad (1)$$

模型中, Cost 是测试成本, t 表示测试时间, tsv 是用于测试的硅通孔数目, α 是影响因子,取值在 (0, 1) 之间,这个值可以由测试工程师设定,用来权衡测试时间和 TSV 数目对测试成本影响的大小.由于在实际工程中,TSV 个数对测试成本的影响也是非常大的,所以 α 在设定的时候会非常小,以便于平衡 TSV 和测试时间对最终成本的影响.

因此,3D SoC 测试优化策略问题,就是给定三维芯片每一层的测试时间和测试引脚(test pin),用于测试的 TAM 最大带宽(TAM width).需要设计出一种优化的测试策略,通过改变每一层的堆叠次序,以及每一次测试过程中的测试先后顺序,最小化测试成本.

2.2 3D-SoC 针对 TSV 绑定的测试流程

基于 3D SoC 的绑定前和绑定后测试流程如图2所示,本文提出的策略分为两个过程,首先是枚举出该芯片的所有可能的堆叠次序,然后依次选择其中的某一种堆叠次序,运用如图2所示的 TestTimeOptimal 算法对该堆叠次序的 3D SoC 进行基于绑定中测试和绑定后测试的优化,算出需要的测试总时间和用于测试的 TSV 数目,再通过式(1)的测试成本模型算出所需要的总测试成本,接着用同样的策略将剩下来的其他堆叠次序都算出对应的测试成本,最后选出成本最低的芯片堆叠次序.例如针对有三层的 SoC,每一层放置一个核,核用 1 到 3 的数字编号.列出所有可能的堆叠次序,即(1, 2, 3), (1, 3, 2), (2, 1, 3), (2, 3, 1), (3, 1, 2) 和 (3, 2,

1)六种,(1,2,3)表示 SIC1 放在最底层,SIC2 放在次底层,SIC3 放在最上层. 首先针对(1,2,3)堆叠次序进行测试,运用 TestTimeOptimal 算法,最小化该堆叠的测试时间,同时计算出对应的 TSV 数目,最后根据式(1)算出测试的总成本,接下来,剩下的 5 种堆叠次序也是以同样的方式处理,最后,假设得到成本最低的那个堆叠次序是(2,1,3),这个堆叠次序就被选为最佳的堆叠次序,同时给出该次序的测试总成本. 2.3 小节会针对图 2 的流程图中的 TestTimeOptimal 算法进行详细阐述.

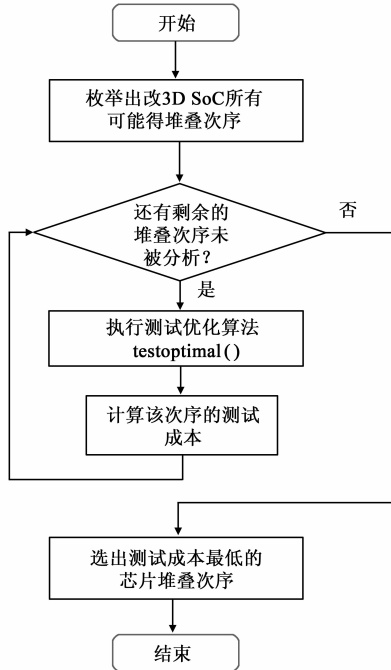


图2 基于3D SoC的绑定前和绑定后测试流程

2.3 3D-SoC 针对 TSV 绑定的测试优化策略

本节详细介绍基于绑定中和绑定后的 3D SoC 测试优化策略. 由于本文基于硬晶体实现的 3D SoC,所以每一层核的信息只包含测试该层的总时间和 TAM 的带宽.

表 1 测试算法的数据结构

| | |
|--------------------|----------------------|
| core_(i). pin | 每一层核的带宽 |
| core_(i). testtime | 每一层核的测试时间 |
| core_(i). end | 该核的测试结束时间,未测试为 -1 |
| core_(i). use | 测试未进行设为 0,已经开始测试设为 1 |
| core_(i). over | 该核已经测试结束设为 1,未结束设为 0 |

数据结构 每一层核的带宽和测试时间会被存放在表 1 的数据结构中. 我们从表 1 中可以看到一个核不仅仅包含了其带宽和测试时间,而且后面的三项是为了算法 1 而设置的.

算法 1 TestTimeOptimal()

输入:

(1)3D SoC 绑定的核的信息,即上文所述的结构体数组集合 C_{core} , corenum 是绑定的核的总数, t, t_1 均是临时时间变量,enum 表示剩余的未被测试的核的个数

(2)TAM_ avail 是提供测试的最大可供带宽

输出:总的测试时间

开始

```

1  初始化  $C_{core}$ 
2  对  $C_{core}$  每一层按照测试时间递减的顺序进行排序,排序之后得到每一层核的集合  $C_{core}'$ 
3   $t := 0$ 
4   $cnum := corenum$ ;
5  WHILE  $cnum \neq 0$  DO
6    FOR core  $i := 0$  to  $corenum$ 
7      IF  $core\_i.pin \leq TAM\_avail$  &&  $core\_i.use = 0$  DO
8         $core\_i.end := core\_i.testtime + t$ 
9         $core\_i.use := 1$ 
10        $cnum := cnum - 1$ 
11     END IF
12   END FOR
13    $t_1 := M // M$  是一个很大的正整数
14   FOR core  $i := 0$  to  $corenum$ 
15     IF ( $t_1 > core\_i.endtime$  &&  $core\_i.endtime \neq 0$  &&  $core\_i.over = 0$ ) DO
16        $t_1 := core\_i.endtime$ ;
17     END IF
18   END FOR
19    $t := t_1$  //将最小的 endtime 给 t
20   选出拥有最小 endtime 的 core_m
21    $core\_m.pin = core\_m.pin + TAM\_avail$ 
22    $core\_m.over = 1$ 
23 END WHILE
24 返回选出集合中 endtime 最大值
结束
    
```

算法 1 适用于 3D SoC 绑定中和绑定后的测试时间的优化计算. 测试的开始时间设为 0,其中第 3 行的变量 t 是一个时间变量,表达了本次循环开始的时间,在此次循环结束之前会改变,然后进入下一次循环. 行 5 ~ 23 是对每一个核进行测试调度,直到所有的核都被测试就结束循环. 行 6 ~ 12 就是按照核测试时间递减的顺序给核分配 TAM 带宽,当 TAM 可供测试的带宽不够的时候,就结束此次循环的分配,将已经分配的最早结束的核的带宽释放,并将最早结束的时间赋给变量 t .

算法 2 是用来计算 3D SoC 绑定中和绑定后的总的测试成本,需要调用算法 1.

算法 2 OverallTestCost()

输入:已经确定堆叠次序的完整的 3D SoC, layernum 是层数,time 是测试所需要的总时间,tsv 是测试所需要的总 TSV 数目

输出:总的测试成本

开始

```

1  t_midbond = 0
2  t_midbond = TestTimeOptimal(0,1) + TestTimeOptimal(0,1,2) +
   ... + TestTimeOptimal(0,1,...,layernum - 2)
3  t_postbond = TestTimeOptimal(0,...,layernum - 1)
4  time = t_midbond + t_postbond
5  tsv = core_(layernum - 1).pin * (layernum - 1) + core_(layernum
   - 1).pin * (layernum - 1) + ... + core_1.pin
6  cost =  $\alpha * t + (1 - \alpha) * tsv$ 
7  return cost
结束
    
```

算法 2 通过调用算法 1,得到了每一次芯片测试中的绑定中测试时间和绑定后测试时间.然后根据本文提出的式(1)的测试成本模型,算出按此种次序堆叠的 3D SoC 测试总成本.由于在 3D SoC 中,测试引脚都是放置在最底层,所以 TAM 需要通过 TSV 来测试上层的核.

图 3 是一个 3 层的三维 SoC 的例子,每一层放置一个核,TAM 支持并行测试,我们从图中可以发现,核 3 在最高层,测试需要通过 TAM3 这条线,而 TAM3 又要穿过中间那一层,所以需要层 3 与层 2 的 TSV,加上层 2 与层 1 的 TSV 个数,即需要两倍的核 3 的测试带宽数,也就是算法 2 针对测试用的 TSV 数目的计算过程.

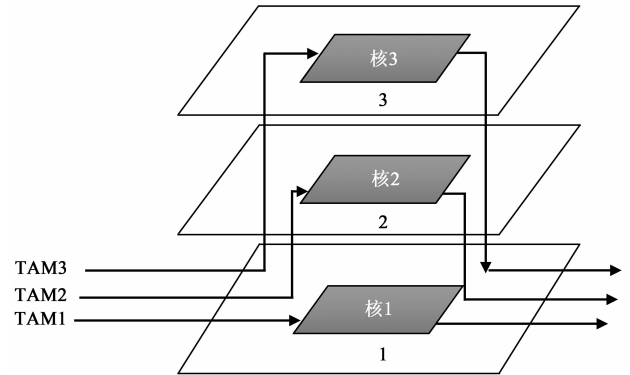


图3 一个3层的三维片上系统的示意图

3 所提的 3D-SoC 测试优化策略评估与分析

为了验证本文提出的 3D SoC 测试优化策略,我们选用了 ITC'02 的基准电路作为被测电路.

如表 2 所示,我们选取的被测电路是具有五层的 3D SoC,这五个核还没有完成堆叠,本文假设每层仅仅放置一个核,表 2 第 3 行的测试时间指的是单独测试该核所需要的时间,第 4 行是指测试该核需要的带宽数.我们可以从表 2 中看出,测试带宽并不完全与测试时间成正比.本文设计的算法是由 C++ 实现的,运行在

2.66GHz 主频的 Intel 处理器上,有 4GB 的内存.所有的程序只需要几毫秒的执行时间.

为了体现我们提出的测试优化策略的优势,我们在表 3 和表 4 中将我们的策略和文献[12]的策略(以下称为基本策略)进行对比.我们通过不断的改变总的 TAM 带宽数,来检测本文的策略在不同环境下的性能体现.

表 2 被测电路信息

| 核 | d695 | t2126 | p22810 | p93791 | p34392 |
|------|--------|--------|---------|---------|---------|
| | 1 | 2 | 3 | 4 | 5 |
| 测试时间 | 106391 | 700665 | 1333098 | 2608870 | 2743317 |
| 测试带宽 | 10 | 20 | 25 | 30 | 25 |

表 3 实验结果对比与分析 ($\alpha = 5 * 10^{-5}$)

| $\alpha = 5 * 10^{-5}$ | 文献[12]提出的方法 | | 本文提出的方法 | | 优化的比例 (P) |
|------------------------|-------------|--------------|--------------|---------|-----------|
| | 最大 TAM 带宽 | 叠放次序 (cost1) | 叠放次序 (cost2) | 成本 | |
| 30 | 1-2-3-4-5 | 333.814 | 4-3-2-5-1 | 296.782 | 11.09% |
| 35 | 1-2-3-4-5 | 333.814 | 4-3-2-5-1 | 296.782 | 11.09% |
| 40 | 1-2-3-4-5 | 333.814 | 4-3-2-5-1 | 296.782 | 11.09% |
| 45 | 1-2-3-4-5 | 323.304 | 4-2-3-5-1 | 286.272 | 11.45% |
| 50 | 1-2-3-5-4 | 316.638 | 4-3-2-5-1 | 271.275 | 14.33% |
| 55 | 1-2-3-4-5 | 302.922 | 4-3-2-5-1 | 245.507 | 18.95% |
| 60 | 1-2-3-4-5 | 302.922 | 4-3-2-5-1 | 245.507 | 18.95% |

表 3 和表 4 是在式(1)取不同 α 值时的实验结果对比,每一次运行本文提出的算法仅需要 5ms 左右的时间.每张表的第 1 列就是测试仪提供的最大 TAM 带宽,第 2 列是采用基本策略所获得的最优的芯片叠放次序,第 3 列是该策略最后的测试总成本.第 4、5 列对应本文提出策略的效果,最后一列优化的比例 P 是由式(2)计算得出的.

表 4 实验结果对比与分析 ($\alpha = 1 * 10^{-5}$)

| $\alpha = 1 * 10^{-5}$ | 文献[12]提出的方法 | | 本文提出的方法 | | 优化的比例 (P) |
|------------------------|-------------|--------------|--------------|---------|-----------|
| | 最大 TAM 带宽 | 叠放次序 (cost1) | 叠放次序 (cost2) | 成本 | |
| 30 | 1-2-3-4-5 | 407.628 | 3-2-1-4-5 | 390.959 | 4.09% |
| 35 | 1-2-3-4-5 | 407.628 | 3-2-1-4-5 | 390.959 | 4.09% |
| 40 | 1-2-3-4-5 | 407.628 | 3-2-1-4-5 | 390.959 | 4.09% |
| 45 | 1-2-3-4-5 | 386.608 | 3-2-1-4-5 | 362.932 | 6.12% |
| 50 | 1-2-3-5-4 | 373.277 | 3-2-5-1-4 | 341.717 | 8.45% |
| 55 | 1-2-3-4-5 | 345.843 | 4-3-2-5-1 | 311.015 | 10.07% |
| 60 | 1-2-3-4-5 | 345.843 | 4-3-2-5-1 | 311.015 | 10.07% |

$$P = (\text{cost1} - \text{cost2}) / \text{cost1} \quad (2)$$

由式(2)可以看出,P 是用来表示本文的方法较基本策略降低了更多成本的百分比.

通过分析两张表的结果,我们观察到的第一个现象是,本文提出的测试策略较基本策略都有所提高,随着 TAM 最大带宽越来越大,本文提出的策略不仅获得了更低的测试总成本,且优化的效果越来越好,表 3 中本文的策略比基础策略最高多降低了 18.95% 的测试

成本,而表 4 也最高多降低了 10.07% 的测试成本。

第二个分析现象是 α 的取值对最终的对比效果有影响,之所以将 α 取很小的值,是因为式(1)中测试时间较 TSV 数目大了几个数量级,这样取值可以平衡两者对最终测试成本的影响。

4 结论

本文提出了基于 3D SoC 绑定中和绑定后的测试策略,这种策略结合了改变芯片堆叠的次序,可以有效的降低测试时间。由于测试时间和用于测试的硅通孔(TSV)数目都对最终的测试成本有很大的影响,所以本文提出了一种测试成本模型,可以同时考虑以上两者的影响,最终达到了最大化的降低测试总成本的目的。实验结果证明我们提出的测试策略较同类的策略,可以更好的降低测试成本。

参考文献

- [1] P GARROU, et al. Handbook of 3D Integration: Volume 1-Technology and Applications of 3D Integrated Circuits[M]. New York: John Wiley & Sons, 2011.
- [2] ERIC BEYNE, et al. 3D system integration technologies[A]. IEEE International Conference on Integrated Circuit Design and Technology [C]. Washington, DC, USA: IEEE Computer Society, 2007. 1-3.
- [3] B BLACK, et al. 3D processing technology and its impact on IA32 microprocessors[A]. Proceedings of the Computer Design VLSI in Computers and Processors [C]. Washington, DC, USA: IEEE Computer Society, 2004. 316-318.
- [4] XIE Y, et al. Design space exploration for 3D integrated circuits[J]. ACM Journal on Emerging Technologies in Computing Systems, 2006, 2(2): 65-103.
- [5] T VUCUREVICH. The Long Road to 3-D Integration: Are We There Yet? [R]. CA, USA: Cadence Berkeley Research Lab, 2007.
- [6] 韩银河, 张磊, 李晓维. 三维芯片的测试技术研究进展[J]. 信息技术快报, 2010, 8(2): 29-35.
HAN Yin-he, ZHANG Lei, LI Xiao-wei. Three-dimensional chip testing technology research[J]. Information Technology Letter, 2010, 8(2): 29-35. (in Chinese)
- [7] EJ MARINISSEN. A structured and scalable mechanism for test access to embedded reusable core[A]. Proceedings of the International Test Conference [C]. Washington, DC, USA: IEEE Computer Society, 1998. 284-293.
- [8] EJ MARINISSEN, et al. Testing 3D chips containing through-silicon vias[A]. Proceedings of the International Test Conference [C]. Washington, DC, USA: IEEE Computer Society, 2009. 1-11.
- [9] GOEL S K, et al. Layout-driven SoC test architecture design for test time and wire length minimization[A]. Proceedings of Design, Automation and Test in Europe Conference and Exhibition [C]. Washington, DC, USA: IEEE Computer Society, 2003. 738-743.
- [10] E LARSSON, et al. A reconfigurable power-conscious core wrapper and its application to SoC test scheduling. Proceedings of Design [A]. Automation and Test in Europe Conference and Exhibition [C]. Washington, DC, USA: IEEE Computer Society, 2003. 1135-1144.
- [11] B NOIA, et al. Design-for-Test and Test Optimization Techniques for TSV-based 3D Stacked ICs [M]. New York: Springer, 2014. 181-237.
- [12] M PRADHAN, et al. Optimal stacking of SoCs in a 3D-SIC for post-bond testing[A]. Proceedings of the 3D Systems Integration Conference [C]. Washington, DC, USA: IEEE Computer Society, 2013. 1-5.
- [13] M TAOUIL, et al. Test impact on the overall die-to-wafer 3D stacked IC cost [J]. Journal of Electronic Testing, 2012, 28(1): 15-25.

作者简介



神克乐 男, 1988 年 1 月出生于江苏省南京市。现为清华大学计算机系博士生。主要研究方向是三维集成电路测试。

E-mail: sk110@mails.tsinghua.edu.cn



虞志刚 男, 1989 年生于安徽省宿松县。现为清华大学计算机系博士生。研究方向为并行与分布式计算、片上网络路由。

E-mail: yuzg@live.com



白宇 男, 1992 年生于河北省衡水市。现为清华大学软件学院硕士生, 研究方向为 VLSI 测试。

E-mail: hbbaiyu@126.com